
DIPLOMARBEIT

Herr Ing. Herbert Dirnberger

**Entwurf einer Steuerung und
Visualisierung für eine Zug-
prüfmaschine**

Ainring, 2016

DIPLOMARBEIT

Entwurf einer Steuerung und Visualisierung für eine Zug- prüfmaschine

Autor:

Herr Ing. Herbert Dirnberger

Studiengang:

Technische Informatik

Seminargruppe:

KT11WIA-F

Hochschulbetreuer:

Herr Prof. Dr.-Ing. Olaf Hagenbruch

Betrieblicher Betreuer:

Herr Dipl.-Ing. Karl-Heinz Schönamsgruber

Einreichung:

Jänner 2016

Bibliografische Angaben:

Dirnberger, Herbert:

Entwurf einer Steuerung und Visualisierung für eine Zugprüfmaschine,

Mittweida, Hochschule Mittweida (FH),

Fachbereich Elektro- und Informationstechnik, Diplomarbeit, 2016

Referat:

Die Arbeit behandelt die Entwicklung der Steuerungstechnik für eine Zugprüfmaschine. Nach einer Einleitung werden zunächst die Hardwarebausteine des Systems erläutert. Danach wird der Aufbau der Steuerungssoftware erklärt. Anschließend wird das Visualisierungskonzept vorgestellt. Zum Schluss folgen eine Zusammenfassung sowie ein Ausblick auf zukünftige Erweiterungen.

Inhalt

Inhalt	i
Abbildungsverzeichnis	iv
Tabellenverzeichnis	vi
Abkürzungsverzeichnis	vii
1 Einleitung.....	9
1.1 Überblick.....	9
1.2 Motivation	9
1.3 Aktueller Stand.....	10
1.4 Aufgabenstellung	12
1.4.1 Prinzipielle Messanordnung	12
1.4.2 Mechanische Konstruktion	14
2 Hardware.....	15
2.1 Zentrale Steuerung	15
2.2 Horizontalantrieb und Vertikalantrieb.....	16
2.3 Probenheizung.....	17
2.4 Heißluftheizung	18
2.5 Kraftmesszelle	19
2.6 Temperatursensoren.....	20
2.7 SPS Übersicht.....	20
3 Software.....	22
3.1 Tools / IDE	22
3.2 Programmaufbau	23
3.3 Schrittkette für den Messablauf.....	25
3.4 Auswertung des Kraftsensors.....	27
3.4.1 Ansteuerung der Messelektronik	27
3.4.2 Auswertung des Kraftsensors	29
3.5 Die Linearantriebe.....	31

3.5.1	Ansteuerung	31
3.5.2	Inbetriebnahme der Achsen	33
3.5.3	Kraftbegrenzung	33
3.6	Probenheizung.....	36
3.7	Heißluftheizung.....	36
3.8	Aufzeichnung der Messwerte.....	37
4	Visualisierung	39
4.1	Auswahl der Visualisierungs-Plattform	39
4.2	Grundlagen Web-Programmierung	41
4.2.1	HTML.....	41
4.2.2	CSS	42
4.2.3	JavaScript.....	42
4.2.4	Webseite.....	43
4.3	Das Tool Web2PLC	44
4.3.1	Arbeitsschritte	45
4.3.2	Einstellungen in Web2PLC	45
4.3.3	AWP-Kommandos	46
4.3.3.1	SPS-Variable lesen.....	46
4.3.3.2	Prozesswerte zyklisch aktualisieren.....	47
4.3.3.3	SPS-Variablen schreiben.....	47
4.3.4	Animierte Grafiken	49
4.3.4.1	SVG-Element.....	49
4.3.4.2	Canvas-Element	50
4.3.5	Tools für die Web-Programmierung	50
4.4	Aufbau der Benutzeroberfläche.....	52
4.4.1	Startbild	52
4.4.2	Einstellungen	53
4.4.3	Messung, Istwerte.....	55
4.4.4	Ausgabe in Tabellenformat	56
4.4.5	Handbetrieb	57
4.4.6	Störungsanzeige.....	58
4.4.7	Justieren des Kraftsensors.....	59
5	Ergebnis	60
5.1	Zusammenfassung	60
5.2	Ausblick	61
5.3	Fazit.....	61
	Literatur	63

Anlagen, Teil 1: Technische Daten Kraftsensor.....	65
Anlagen, Teil 2: Technische Daten Siwarex	66
Anlagen, Teil 3: Befehlscodes Siwarex CS.....	67
Anlagen, Teil 4: Bilder der Zugprüfmaschine.....	69
Eidesstattliche Erklärung	70

Abbildungsverzeichnis

Abbildung 1 - Schnitt durch ein kaschiertes Bauteil [FrHoe2013] S. 5	10
Abbildung 2 - Manueller Zugversuch [FrFrau2013] S. 4	11
Abbildung 3 - Skizze prinzipieller Aufbau [FrFrau2013] S. 3.....	12
Abbildung 4 - Gewünschter Messablauf	13
Abbildung 5 - Unerwünschte Effekte	13
Abbildung 6 - CAD-Konstruktion [DinKo2014] S. 17	14
Abbildung 7 - Linearachse, Servomotor und Motorcontroller [FesSp2015] S. 5, S. 8	16
Abbildung 8 - flexibler Rohrheizkörper [Hotset2013].....	17
Abbildung 9 – Luftherhitzer [Leister2013].....	18
Abbildung 10 – Kraftmesszelle [Me2014]	19
Abbildung 11 - SPS Übersicht.....	20
Abbildung 12 - Programmstruktur.....	24
Abbildung 13 - Messablauf.....	26
Abbildung 14 - Abhängigkeiten der Schnittstellenbausteine	28
Abbildung 15 - Spannungs-Dehnungs-Diagramm [Bon2014] S.62.....	29
Abbildung 16 - Abreißerkennung.....	30
Abbildung 17 - Abreißerkennung Algorithmus	31
Abbildung 18 - Kommunikationsmöglichkeiten der Achsen [Fhpp2009] S. 1-3	32
Abbildung 19 - Konfigurationstool der Achsen.....	33
Abbildung 20 - Formel Vorschubkraft Spindeltrieb [Thk2006] S. 15-7	34

Abbildung 21 - Kraftregelung	35
Abbildung 22 - Regelung Heißlufttheizung	37
Abbildung 23 - Schnittstelle Datenlogger.....	38
Abbildung 24 - Beispiel eines einzelnen HTML-Elementes.....	41
Abbildung 25 - Beispiel Verschachtelte HTML-Elemente.....	41
Abbildung 26 - Beispiel CSS-Anweisung	42
Abbildung 27 - Beispiel zur JavaScript-Syntax	43
Abbildung 28 - Webserver und Webbrowser	43
Abbildung 29 - S7-Web2PLC	44
Abbildung 30 - Verzeichnisse	45
Abbildung 31 - Beispiel SPS-Variable lesen	46
Abbildung 32 - Einmalige und zyklische Aktualisierung	47
Abbildung 33 - Beispiel SPS-Variable schreiben	48
Abbildung 34 - Beispiel Variable setzen über Schaltfläche	49
Abbildung 35 - Netzwerk Profiler	51
Abbildung 36 - Startbild	52
Abbildung 37 - Einstellungen.....	53
Abbildung 38 - Messung.....	55
Abbildung 39 - Tabelle	56
Abbildung 40 - Handbetrieb.....	57
Abbildung 41 - Störmeldungsanzeige.....	58
Abbildung 42 - Justieren.....	59

Tabellenverzeichnis

Tabelle 1 - Vergleich der Visualisierungsmethoden.....	40
Tabelle 2 - Pro und Contra der Visualisierungsmethoden	40

Abkürzungsverzeichnis

SPS	Speicherprogrammierbare Steuerung
HMI	Human Machine Interface
SCADA	Supervisory Control and Data Acquisition
CPU	Central Processing Unit
F&E	Forschung und Entwicklung
TCP/IP	Transmission Control Protocol / Internet Protocol
CAD	computer-aided design
IDE	integrated developer environment
SSR	Solid State Relais
DO	Digital Output
DI	Digital Input
AO	Analog Output
AI	Analog Input
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
DMS	Dehnungsmessstreifen
KOP	Kontaktplan
FUP	Funktionsplan
AWL	Anweisungsliste
SCL	Structured Control Language
SFC	Sequential Function Chart
FC	Funktion
FB	Funktionsbaustein
DB	Datenbaustein
E/A	Eingang / Ausgang
HW	Hardware
FHPP	Festo Handling and Positoning Protocol
DOM	Document Object Model
AJAX	Asynchronous JavaScript and XML
AWP	Anwender Webseiten Programmierung

1 Einleitung

1.1 Überblick

Diese Arbeit wurde in Zusammenarbeit mit der Firma Frimo Control Systems GmbH erstellt. Frimo stellt Maschinen für die Kunststoffverarbeitung in der Automobilindustrie her. Ein großes Standbein stellen die sogenannten Kaschieranlagen dar. Sie dienen zum Veredeln von Kunststoffteilen. Dabei werden die sonst glatten Kunststoffteile mit einer weiteren Schicht aus Kunstleder oder Echtleder überzogen. Die kaschierten Bauteile müssten strengen Qualitätsanforderungen genügen. Neben der unversehrten Optik ist auch die ausreichende und dauerhafte Haftung der Deckschicht auf dem Kunststoffteil von entscheidender Bedeutung für die Automobilhersteller.

1.2 Motivation

Es ist ebenfalls im Interesse des Kunden, dass die zur Verfügung stehende Produktionszeit möglichst gut ausgenutzt wird. Höhere Produktionsraten führen zu einer schnelleren Amortisierung der Investitionskosten und somit zu einem günstigeren Produktpreis. Aus diesem Grund versucht die Firma Frimo die Fertigungszeit ihrer Anlagen so kurz wie möglich zu halten. Dieses Ausreizen führt aber in vielen Fällen zu Problemen bei der Produktqualität.

Um diesen Problemen zu begegnen entschloss sich die Abteilung F&E (Forschung und Entwicklung) eine Zugprüfmaschine anzuschaffen. Sie soll die Haltekräfte der Deckschicht in verschiedenen Situationen messen. Durch sukzessive Anpassung der Prozessparameter an den Anlagen und anschließende Messung der Haltekräfte können so die optimalen Einstellungen für die verschiedenen Materialkombinationen ermittelt und dokumentiert werden.

Diese Parameter sind u.a.:

- Aufheizzeit des Klebers bis zur Aktivierungstemperatur
- Heizleistung
- Menge des Klebers
- Klebermischung, Klebertyp, Kleberhersteller
- Abkühlzeit bis zur Aushärtetemperatur
- Temperatur bei der das Bauteil freigegeben wird
- Anpresskraft und Presszeit

Kommerziell erhältliche Zugprüfmaschinen sind teuer und enthalten nicht die speziellen Features, die für die vorliegenden Aufgaben gebraucht werden. Es handelt sich hier nicht um einen klassischen Zugversuch, bei dem ein einteiliger Probenkörper auseinandergerissen wird. Es geht vielmehr um einen zweischichtigen Probenkörper, von dem eine

Schicht abgeschält wird. Da man die Zugprüfmaschine nach eigenen Vorstellungen gestalten und für zukünftige Erweiterungen nicht von einem externen Zulieferer abhängig sein wollte, wurde beschlossen die Maschine durch die in der Firma vorhandenen Kapazitäten selbst anzufertigen.

Die mechanische Konstruktion wurde ebenfalls im Rahmen einer weiteren Diplomarbeit [DinKo2014] erstellt. Diese wurde dann im Jahr 2015 zur Fertigung freigegeben. Die vorliegende Arbeit beschäftigt sich nun mit der steuerungstechnischen Ausrüstung und der Programmierung dieser Zugprüfmaschine.

1.3 Aktueller Stand

Bild 1 zeigt einen Schnitt durch ein Kaschiertes Bauteil. Es sind deutlich die verschiedenen Schichten zu sehen deren Haltekräfte ermittelt werden sollen.

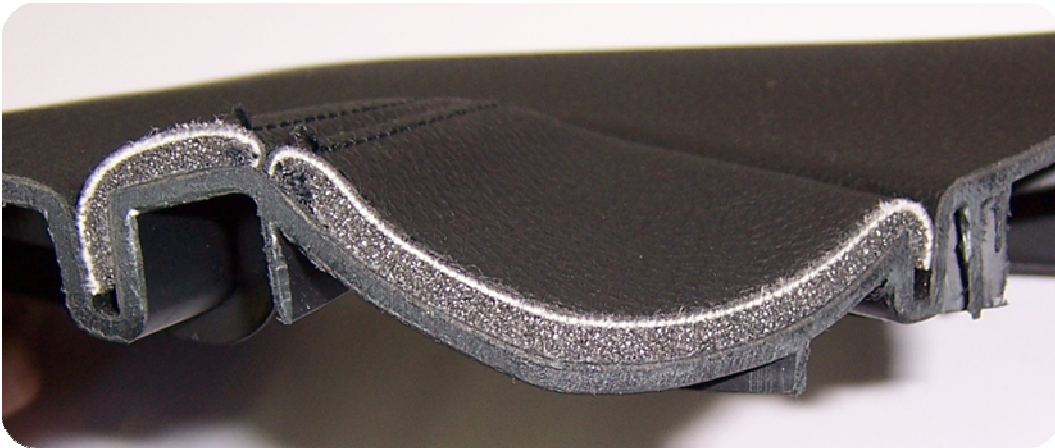


Abbildung 1 - Schnitt durch ein kaschiertes Bauteil [FrHoe2013] S. 5

Für die Messungen werden rechteckige Proben angefertigt, welche aus demselben Material bestehen wie das Endprodukt. Diese Proben werden dann unter verschiedenen Umgebungsbedingungen kaschiert. In der Zugprüfmaschine wird dann die Deckschicht wieder abgezogen und die dafür erforderlichen Kräfte gemessen. Im Normalfall bleibt die Deckschicht dabei intakt, man spricht auch vom „Abschälen“. Nachfolgendes Bild soll das verdeutlichen.

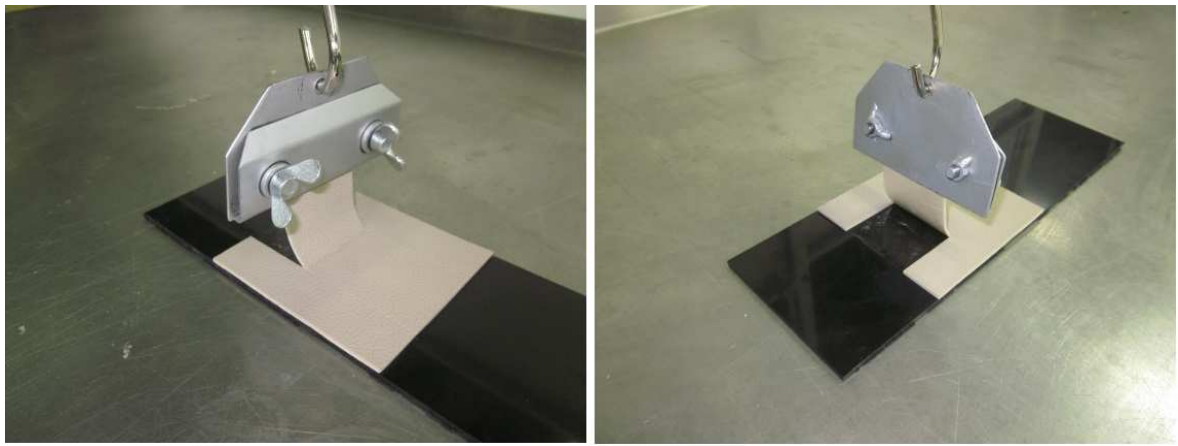


Abbildung 2 - Manueller Zugversuch [FrFrau2013] S. 4

Bisher wurden solche Versuche oft manuell durchgeführt. Das Abziehen erfolgte durch Muskelkraft. Als Messgerät verwendete man eine Federzugwaage. Diese Methode dient eher der groben Evaluation und lässt viel Spielraum offen. Wichtige Rahmenbedingungen, wie z.B. die Abreißgeschwindigkeit, der Abreiß-Winkel und die Umgebungstemperatur, bleiben unberücksichtigt. Die so durchgeführten Messungen sind untereinander kaum vergleichbar. Die Zugprüfmaschine verspricht hier eine deutliche Verbesserung.

1.4 Aufgabenstellung

1.4.1 Prinzipielle Messanordnung

Im Folgenden werden nun der Aufbau und die Funktionsweise der Zugprüfmaschine erläutert. Abbildung 3 zeigt die wichtigsten Bestandteile:

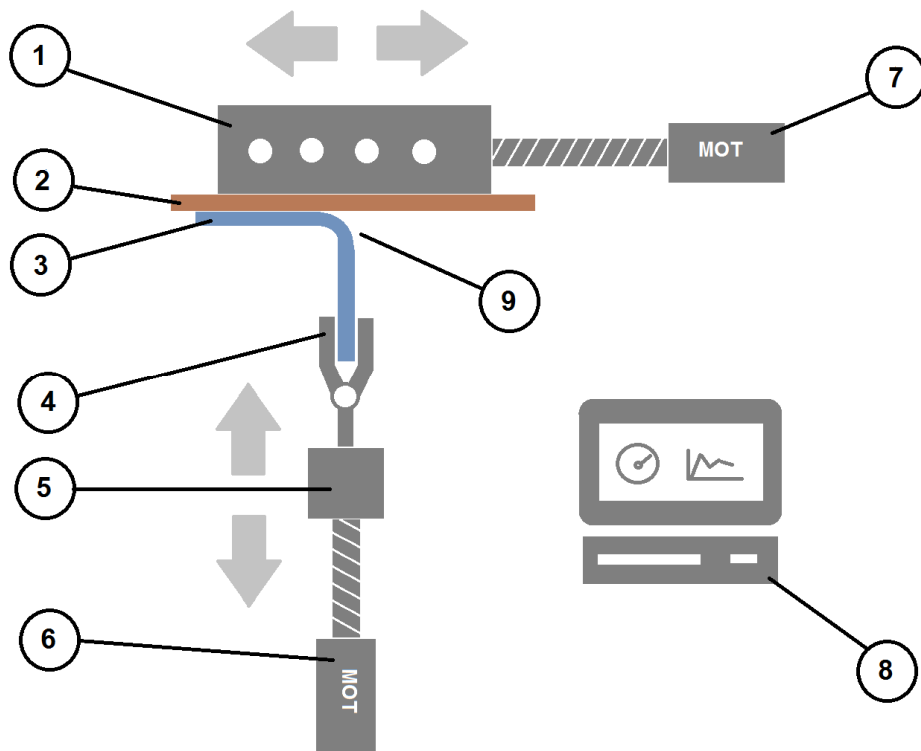


Abbildung 3 - Skizze prinzipieller Aufbau [FrFrau2013] S. 3

Die Probenaufnahme (1) verfügt über Kanäle für eine elektrische Heizung. Dadurch kann die Probe über die Raumtemperatur aufgeheizt und konstant auf einer bestimmten Temperatur gehalten werden. Zusätzlich gibt es eine geregelte Heißluftheizung, welche für eine definierte Lufttemperatur im Probenraum sorgt. Ohne diese würde die Probe nur von der Rückseite erwärmt werden und es würde sich ein Temperaturgefälle im Material einstellen. Die Probe, bestehend aus Trägermaterial (2) und Deckschicht (3), wird in die Probenaufnahme (1) eingespannt. Die Deckschicht wird in die Spannzange (4) eingespannt. Der Vertikaltrieb (6) zieht die Deckschicht vom Trägermaterial ab, während die Kraftmesszelle (5) die Abreißkräfte aufnimmt. Dabei wandert die Abreißkante (9) von rechts nach links und würde somit den Abreißwinkel ändern. Um diesen Effekt zu kompensieren wird die Probenaufnahme durch den Horizontaltrieb (7) gleichzeitig von links nach rechts bewegt. Somit ergibt sich während der Messung ein gleichbleibender Abreißwinkel.

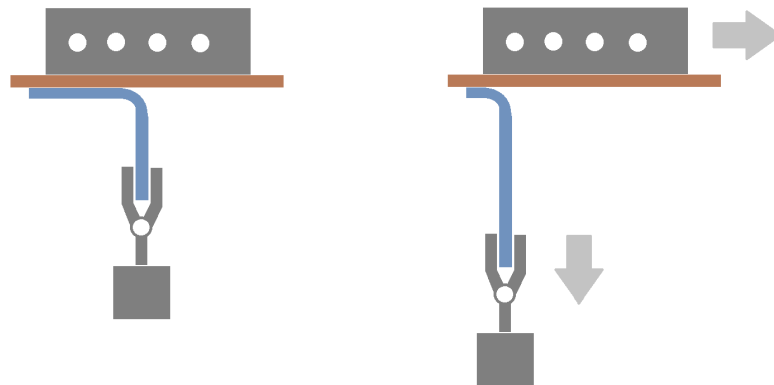


Abbildung 4 - Gewünschter Messablauf

Abbildung 4 zeigt den gewünschten Ablauf. Links ist die Ausgangssituation dargestellt. Startet der Zugversuch, dann muss zuerst der Vertikalantrieb auf Spannung gehen und dann erst der Horizontalantrieb zuschalten. So ergibt sich stets ein gleichbleibender Abreißwinkel von 90°.

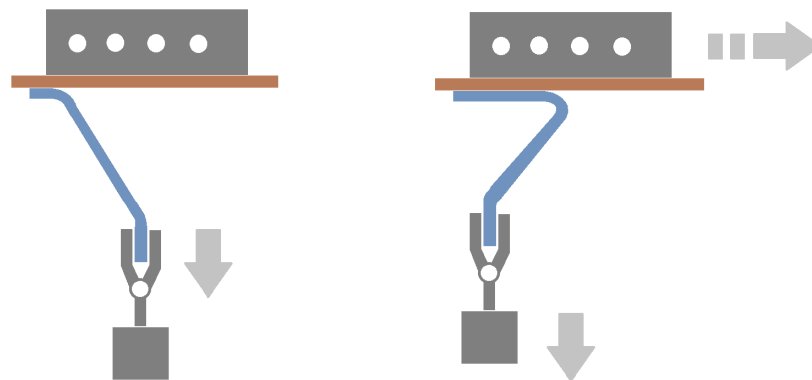


Abbildung 5 - Unerwünschte Effekte

Hier sind zwei unerwünschte Situationen dargestellt. Links die Situation wenn der Horizontalantrieb nicht aktiv wird. Der Vertikalantrieb reißt die Deckschicht ab, während die Abreißkante nach links wandert. Dies würde die Messergebnisse verfälschen und ist daher unerwünscht. Auf der rechten Seite die Situation wenn der Horizontalantrieb zu früh losfährt. Hier wird die Abreißkante nach rechts verschoben, was ebenfalls zu einem Veränderten Messergebnis führen würde.

1.4.2 Mechanische Konstruktion

Das Konstruktionsteam entwickelte auf Basis des Lastenheftes einen CAD-Entwurf, welcher hier kurz vorgestellt wird.

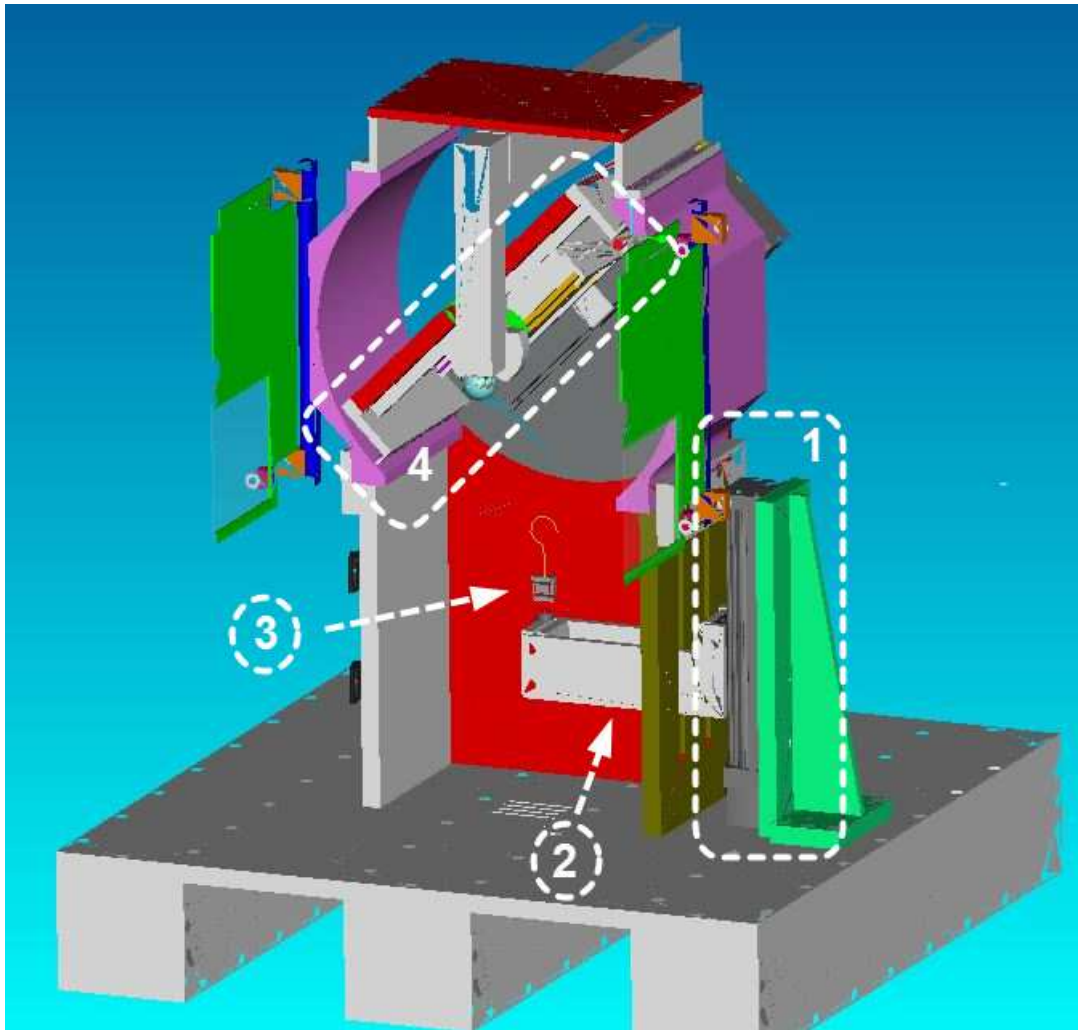


Abbildung 6 - CAD-Konstruktion [DinKo2014] S. 17

Die Messkammer ist von einem Gehäuse umschlossen damit keine kalte Umgebungsluft an die Probe gelangen kann. Die Vorderseite ist über Scharniere wie eine Tür aufklappbar. Ein regelbares Heißluftgebläse spült die Kammer mit temperierter Luft. Der Vertikal-antrieb (1) ist auf der rechten Seite außerhalb des Gehäuses angebracht. Ein Arm aus Plattenmaterial (2) ragt durch Schlitz hindurch in die Messkammer. Auf dem Arm ist der Kraftsensor (3) montiert. Ein Haken dient zum Einhängen verschiedener Zangen um das Probenende einzuspannen. Im oberen Bereich befindet sich der Horizontalantrieb (4) mit der Probenaufnahme, hier in 45°-Stellung. Der gesamte obere Bereich drehbar gelagert. Damit kann die Probenaufnahme samt Horizontalantrieb um ± 45 Grad schräg gestellt werden. Somit können verschiedene Abzugswinkel getestet werden. Die Probe wird Kopfüber eingespannt. Die Deckschicht wird nach unten hin abgezogen. Für diese Maschine musste nun die passende Steuerungstechnik entwickelt werden. Das ist Thema der vorliegenden Arbeit.

2 Hardware

Dieser Abschnitt beschreibt die wichtigsten Hardwarekomponenten sowie deren Schnittstellen. In einem Gespräch mit der Abteilung F&E wurden folgende Anforderungen an die Hardware definiert:

- Temperaturregelung 30 bis 80 °C, Genauigkeit +/- 1 °C
- Vorschubgeschwindigkeit 0,1 bis 2 mm/s
- Vorschubweg 0 bis 200 mm, Genauigkeit +/- 0,05 mm
- Vorschubkraft 5 bis 50 N, Genauigkeit +/- 0,5 N

2.1 Zentrale Steuerung

In der Automobilindustrie sind die Steuerungen der Firma Siemens sehr weit verbreitet. So hat sich auch die Firma Frimo für speicherprogrammierbare Steuerungen (SPS) von Siemens entschieden. Die Lizenzen für die Programmierumgebung Step7 sowie Kenntnisse der Programmiersprache AWL (Anweisungsliste) sind daher bereits im Unternehmen vorhanden. Es ist deswegen naheliegend auch im vorliegenden Projekt eine Step7-Steuerung einzusetzen. Die Wahl fiel auf die Steuerung vom Typ IM151-8 F-CPU aus der Baureihe ET200S. Der Grund ist einerseits das gute Preis-Leistungsverhältnis, andererseits die gute Ausstattung. Wegen der ständig steigenden Rechenleistung sowie der unkomplizierten Erweiterbarkeit verwenden einige Hersteller nicht mehr den Begriff „speicherprogrammierbare Steuerung“ sondern nennen ihre Produkte stattdessen „modulare Controller“. In dieser Arbeit wird jedoch weiterhin der Begriff SPS verwendet.

Ausgewählte Technische Daten:

- Arbeitsspeicher 192 kB
- Flash 1 MB, durch Steckkarte erweiterbar
- Rechenzeiten: Bitoperation 0.06 µs, Festpunkoperation 0.12 µs, Gleitpunkttop. 0.59 µs

Kommunikationsmöglichkeiten:

- Rückwandbus zur Kommunikation mit Herstellereigenen E/A-Modulen
- Industrial Ethernet über drei Switch-Ports
- Protokolle: Profinet-IO, TCP, UDP, ISO-on-TCP, Webserver, S7-Kommunikation

2.2 Horizontalantrieb und Vertikalantrieb

In der mechanischen Konstruktion wurden zwei Linearachsen mit Spindelantrieb von der Firma Festo vorgesehen. Da dieser Hersteller auch die dazu passende Antriebselektronik zur Verfügung stellt fiel die Wahl auf eben diesen. Allerdings machen die geforderten geringen Geschwindigkeiten von 0,1 mm/s bis 2 mm/s ein hohes Übersetzungsverhältnis notwendig. In Absprache mit der Fa. Festo konnte nach einiger Zeit ein passendes Getriebe gefunden werden.

Für die Dimensionierung der Komponenten stellt Festo eigens die Auslegungssoftware „PositioningDrives“ zur Verfügung. Es beinhaltet eine Datenbank mit den wichtigsten Produktdaten. Durch Eingabe von Eckdaten, wie z.B. bewegte Masse und gewünschte Beschleunigung, wird der jeweilige Anwendungsfall beschrieben. Die Auslegungssoftware schlägt darauf hin verschiedene Produktkombinationen vor. Die endgültige Auswahl erfordert jedoch ein gewisses Maß an Erfahrung und erfolgt in der Regel in Zusammenarbeit mit der Herstellerfirma.

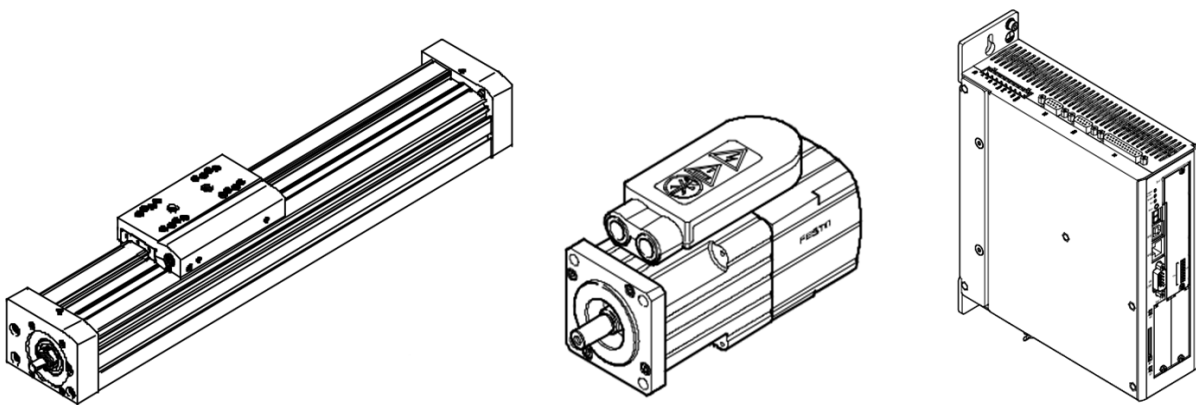


Abbildung 7 - Linearachse, Servomotor und Motorcontroller [FesSp2015] S. 5, S. 8

Abbildung 7 zeigt die in der Zugprüfmaschine eingesetzten Antriebskomponenten. Links die Linearführung mit eingebauter Kugelumlaufspindel. Sie erträgt eine Vorschubkraft von 300 N und weist eine Wiederholgenauigkeit von $\pm 0,02$ mm auf [FesBa2015]. In der Mitte befindet sich der Servomotor. Es handelt sich um einen kompakten permanenterregten Synchron-Servomotor inklusive Absolutwertgeber. Der Geber löst eine Motorumdrehung in 18 Bit auf, was mehr als genug für die geforderte Genauigkeit ist. Weitere 12 Bit werden für den Umdrehungszähler verwendet. Rechts ist die Leistungselektronik dargestellt. Es handelt sich um einen intelligenten Servocontroller. Er kann verschiedene Tätigkeiten, wie z.B. Positionieren und Satzsteuerung, selbstständig ausführen. Er ist durch seinen Funktionsumfang universell einsetzbar. Es stehen drei Einschubschächte zur Verfügung, mit denen der Servocontroller erweitert werden kann. Ein Schacht wurde mit einer Profinet-Kommunikationskarte bestückt, über die die Kommunikation mit der SPS abläuft. Weitere Technische Daten sind im Anhang zu finden.

2.3 Probenheizung

Die Probenaufnahme besteht aus einem Quader aus Aluminium in den eine Schlangenförmige Nut gefräst wurde. Die Nut ist für die Aufnahme einer Heizschlange vorgesehen.

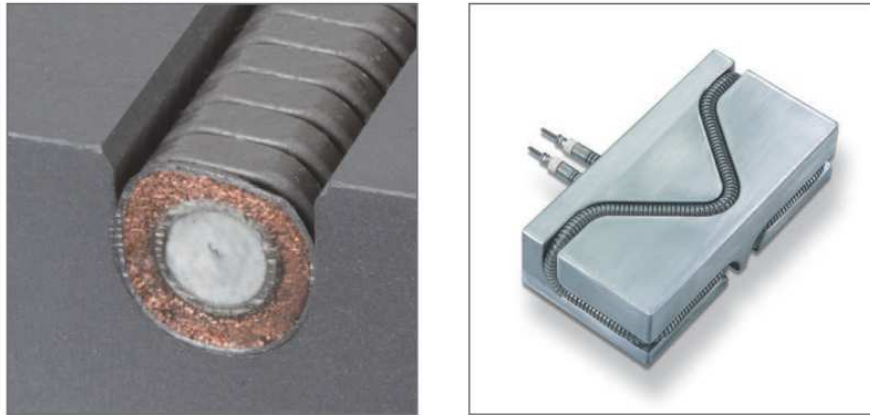


Abbildung 8 - flexibler Rohrheizkörper [Hotset2013]

Die Probenheizung hat eine relativ hohe Leistung von 1000 Watt. Zur Leistungsregulierung dient ein SSR (Solid State Relais). Das SSR wird durch einen Softwareregler mit Pulsweitenmodulation angesteuert. Zusammen mit einem Temperatursensor kann somit ein typischer Regelkreis aufgebaut werden. Die Funktion des Reglers übernimmt die zentrale Steuerung. Entsprechende Softwarebausteine stehen zur Verfügung.

Da das SSR nur im Spannungsulldurchgang bzw. im Stromulldurchgang schaltet ergibt sich eine höchst mögliche zeitliche Auflösung von 100 Schaltspielen pro Sekunde. Auf Grund der Tatsache, dass das PWM-Signal per Software erzeugt wird, ist die erreichbare Auflösung ebenfalls von der Bearbeitungsgeschwindigkeit der Software abhängig. In diesem Fall wird der Softwarebaustein im Zyklus von 100ms aufgerufen, wodurch sich eine Auflösung auf 10 Schaltspielen pro Sekunde verschlechtert. Bei einer Periodendauer von einer Sekunde ergibt sich somit eine Auflösung von 100 Watt in zehn Leistungsstufen. Die Praxis zeigt, dass diese Auflösung für einen gleichmäßigen Temperaturverlauf bereits ausreicht. Die hohe Wärmekapazität des Aluminiums führt zu einer großen Zeitkonstante, also zu einer vergleichsweise trägen Regelung. Dies wirkt sich positiv auf das Ergebnis aus. Die Pausen zwischen den Pulsen werden gut überbrückt, am Temperatursensor sind keine Schwankungen erkennbar.

2.4 Heißluftheizung

Die Heißluftheizung dient zur Unterstützung der Probenheizung. Sie spült den Probenraum mit temperierter Luft. Zum Einsatz kommt das Produkt von der Firma Leister. Das Gerät wurde auf Grund seiner guten Ausstattung ausgewählt. Es besitzt bereits ein Gebläse mit variabler Drehzahl und einen integrierten Controller samt Temperaturregelung. Somit muss im Schaltschrank keine zusätzliche Leistungselektronik vorgesehen werden. Sowohl Solltemperatur als auch Luftmenge können über eine Schnittstelle vorgegeben werden. Dabei werden die Sollwerte über standardisierte Analogsignale (4-20mA) von der zentralen Steuerung übergeben.

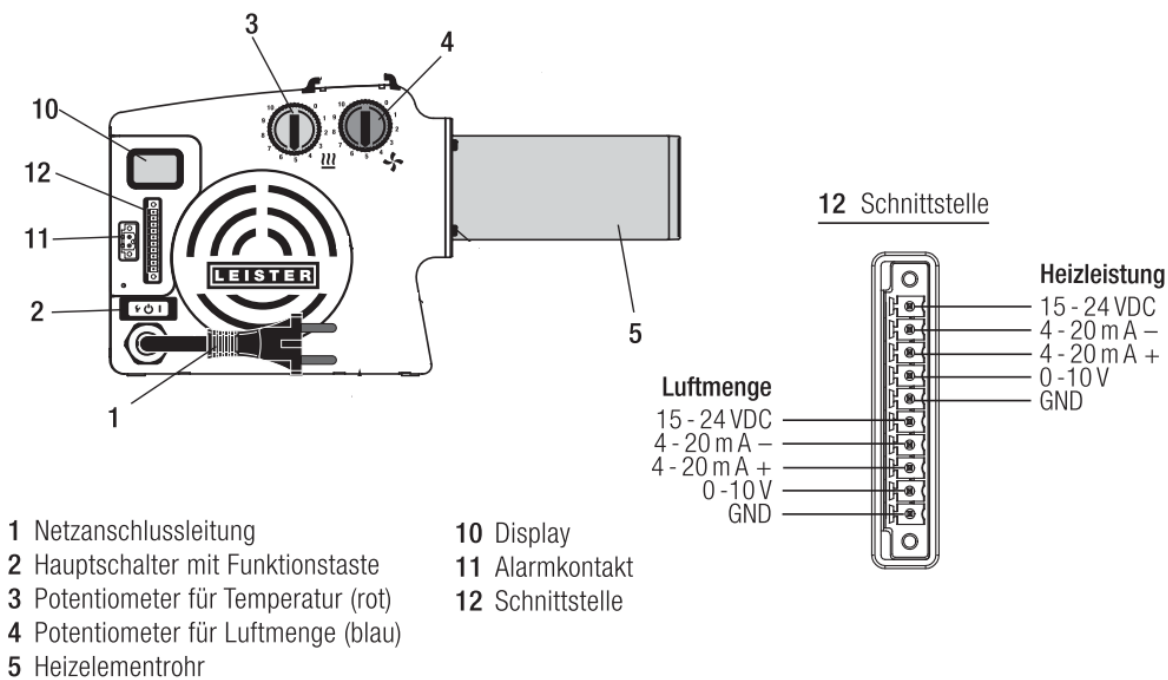


Abbildung 9 – Lufterhitzer [Leister2013]

Das Gerät kann entweder selbständig betrieben werden, indem die Temperatur und Luftmenge über die Drehregler (3) und (4) eingestellt werden, oder Temperatur und Luftmenge werden über die Schnittstelle (12) vorgegeben. Es unterstützt zwei Betriebsarten: Open Loop und Closed Loop. Bei Closed Loop übernimmt der eingebaute Controller selbst die Temperaturregelung, die SPS gibt lediglich die Solltemperatur vor.

2.5 Kraftmesszelle

Eines der wichtigsten Elemente in der Zugprüfmaschine ist der Kraftsensor. Es handelt sich um einen typischen Federkörper-Kraftaufnehmer in S-förmiger Bauart. Die Nennkraft wurde entsprechend den zu erwartenden Abreißkräften mit ± 100 Newton ausgelegt. Die Genauigkeit wird vom Hersteller mit 0,1% vom Messbereichsendwert angegeben. Die Linearitätsabweichung von 0,02 % macht nur einen kleinen Teil aus kann hier vernachlässigt werden.

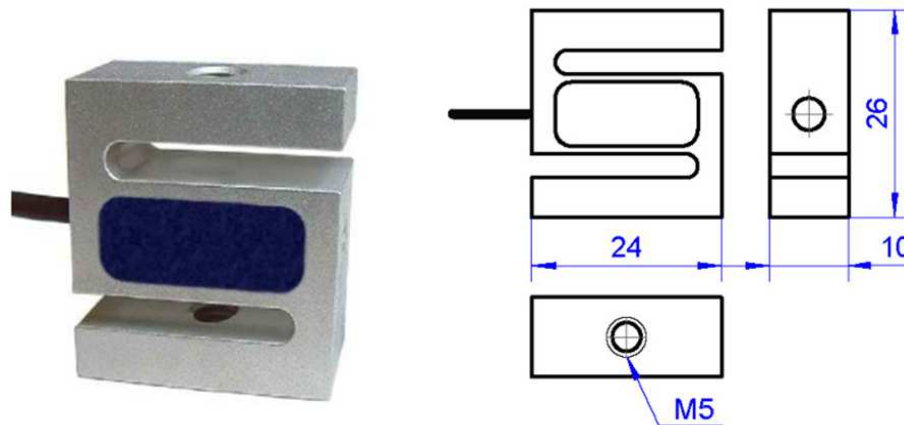


Abbildung 10 – Kraftmesszelle [Me2014]

Im Sensor befinden sich Dehnungsmessstreifen in Brückenschaltung. Idealerweise gibt es speziell hierfür eine passende Auswerteelektronik als SPS-Modul. Das Modul ist für eichpflichtige Wiegeanwendungen zugelassen und wird als E/A-Karte direkt an den Rückwandbus der SPS gesteckt. So kann die DMS-Brücke ohne eine weitere Signalanpassung direkt angeschlossen werden.

Der Messbereich des Modules kann durch die Konfiguration in drei Stufen umgeschaltet werden. In der vorliegenden Anwendung wurde der kleinste Messbereich von 1 mV/V gewählt. Da die Messbrücke im Sensor lediglich ein Signal von 0,5 mV/V ausgibt wird der Wertebereich des AD-Wandlers nie voll angesteuert. Das Datenblatt des SPS-Modules [Siwa2011] deutet auf einen 16-Bit-AD-Wandler hin. Laut [Par2008], S.82 kann der Messfehler in Folge der Quantisierung mit $A=1/2^n$ angenähert werden. Da der negative Wertebereich nicht genutzt wird und der positive Wertebereich nur zur Hälfte genutzt wird, muss die Auflösung auf 14 Bit verringert werden. Obwohl der sich daraus ergebende Messfehler sehr klein ist, wird die Genauigkeit des Modules lediglich mit 0,05% vom Messbereichsendwert angegeben. Der Messbereichsendwert liegt bei 1mV/V und würde 200N entsprechen. Der Messfehler wäre dann 0,1N. Dieser muss zum Fehler des Sensors addiert werden, was zu einem Gesamt-Messfehler von 0,2 Newton führt. Das ist immer noch ein sehr guter Wert und für die vorliegende Aufgabe bestens geeignet.

2.6 Temperatursensoren

Als Temperatursensoren kommen PT100-Sensoren im robusten Industriegehäuse zum Einsatz. Die Genauigkeit wird angegeben mit Klasse B nach DIN EN 60751. Laut [Jum2015], S.2 entspricht das im Temperaturbereich von 0 bis 80°C einem Messfehler von kleiner $\pm 1^\circ\text{C}$. Die Sensoren werden in vier-Leiter-Technik an ein entsprechendes SPS-Eingangsmodul angeschlossen. Das Eingangsmodul löst den Messwert mit einer Granularität von $0,1^\circ\text{C}$ auf. Es kommen drei Temperatursensoren zum Einsatz:

- Lufttemperatur im Innenraum zur Aufzeichnung und zur Regelung des Lufterhitzers.
- Temperatur der Probenaufnahme zur Aufzeichnung und zur Regelung der Heizung
- Gehäusetemperatur, nur zur Anzeige

2.7 SPS Übersicht

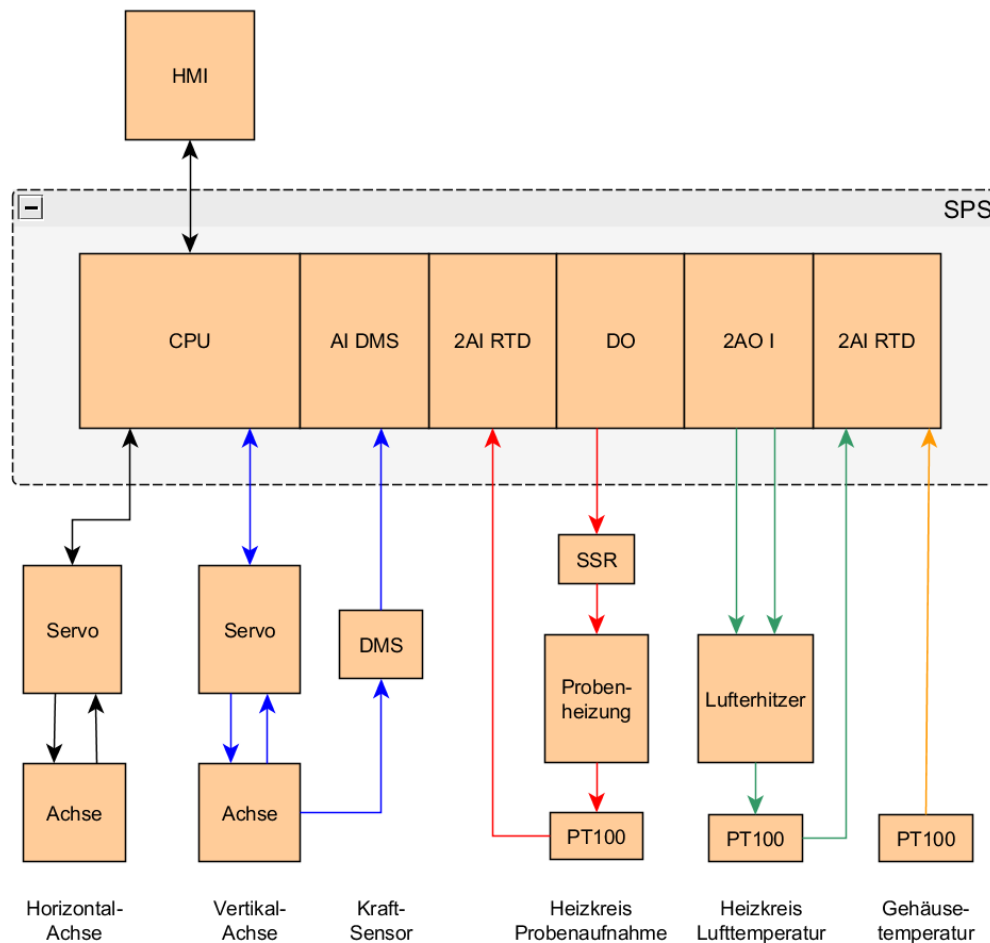


Abbildung 11 - SPS Übersicht

Die Abbildung 11 zeigt die wichtigsten Elemente des Systemes auf einem Bild. Die Pfeile stellen den Informationsfluss dar. In der Mitte der Abbildung befindet sich die SPS. Das Zentrale Bauelement ist das CPU-Modul. Hier passiert die Abarbeitung des SPS-

Programmes und die Speicherung der Messwerte. Die Benutzeroberfläche, auch HMI (Human Machine Interface) genannt, ist über Ethernet mit der CPU verbunden. Sie greift lesend und schreibend auf die Steuerung zu. Die beiden Servocontroller sind über das Feldbussystem Profinet mit der CPU verbunden. Sie regeln die Linearachsen und melden Daten wie Position und Geschwindigkeit an die CPU zurück. Die Vertikalachse verfügt zusätzlich über einen Kraftsensor. Für den Kraftsensor ist ein Spezielles Analogeingangsmodul nötig (AI DMS). Die Verschiedenen E/A-Module werden direkt aneinandergreift, dabei entsteht eine Verbindung zur CPU über den Rückwandbus. Des weiteren sind die Beiden Heizkreise dargestellt. Die Probenheizung wird über ein Solid State Relais angesteuert, dieses wiederum von einem Digital-Ausgangsmodul (DO). Die Temperatur der Probenheizung wird über einen Temperatursensor vom Typ PT100 in die Steuerung zurückgelesen. Dazu wird ein speziell für Widerstandsthermometer vorgesehenes Eingangsmodul verwendet (AI RTD). Ein AI-RTD-Modul besitzt zwei Messkanäle. Weiters ist ein Zweifach-Analogausgangsmodul für Strommodulierte Signale verbaut (2AO I). Es steuert den Lufterhitzer. Die Kanäle arbeiten im standardisierten Bereich von 4-20mA. Ein Kanal steuert das Gebläse, der andere Kanal steuert die Heizleistung. Ein PT100-Temperatursensor gibt die aktuelle Lufttemperatur an die SPS zurück. Ein weiterer Temperatursensor misst die Gegäusetemperatur.

3 Software

Dieses Kapitel behandelt die SPS-Software. Es werden sowohl die verwendeten Tools angesprochen als auch die vom Autor entwickelte Anwender-Software erklärt.

Die SPS-Software muss folgende Kernaufgaben erfüllen:

- Ansteuerung aller Hardwarekomponenten über die jeweiligen Schnittstellen gemäß Herstellerdokumentation
- Einhalten des Messablaufes gemäß Benutzereinstellungen
- Aufnahme der Messwerte

Weitere Features gemäß Kundenwunsch:

- Abreißerkennung über den Kraftsensor
- Justieren des Kraftsensors
- Manuelles Ansteuern der Achsen/Heizung im Handbetrieb
- Probe spannen
- Mehrfach-Messung
- Langzeitmessung

3.1 Tools / IDE

Die Betriebssystem der CPU vom Typ IM151-8F entspricht im Wesentlichen der Baureihe S7-300. Sie wird mit der Entwicklungsumgebung Step7 programmiert. Step7 beinhaltet zahlreiche Tools, welche sowohl das Erstellen des Anwenderprogrammes als auch die Inbetriebnahme und Einstellung adressieren. In der vorliegenden Arbeit wurden folgende Step7-Tools verwendet:

- Simatic Manager V5.5 SP4
- Editor KOP/AWL/FUP
- Tool Bausteinkonsistenz prüfen
- Wizard zur PID-Regleroptimierung
- Editor HW Konfiguration
- S7-Web2PLC
- S7 Referenzdaten

Step7 kennt die folgenden Programmiersprachen:

- KOP (Kontaktplan)
- AWL (Anweisungsliste)
- FUP (Funktionsplan)
- SCL (Strukturierter Text)
- SFC (Ablaufprogrammierung)

Das SPS-Programm der Zugprüfmaschine wurde zur Gänze in der Sprache AWL realisiert, da die Ressourcen des Systems damit effizienter genutzt werden können. Eine Einführung in diese Programmiersprache würde den Rahmen dieser Arbeit sprengen bzw. dürfte vielen Lesern bereits bekannt sein. Stattdessen sei hier auf weiterführende Literatur [WeiZas1998] verwiesen.

3.2 Programmaufbau

Das Betriebssystem der SPS bietet verschiedene Organisationsmöglichkeiten für die Programmaufteilung. Der Programmaufruf kann erfolgen:

- Ereignisgesteuert (Interrupt wenn Prozesswert über-/unterschritten)
- Zeitgesteuert (Zeitinterrupt)
- Freier Zyklus (zyklische Wiederholung des Programmes)

Der Großteil des Programmes wird im freien Zyklus aufgerufen. Die Durchlaufzeit eines Zyklus und damit die Wiederholfrequenz ergeben sich unter anderem aus der Menge und den Verzweigungen der Programmbefehle. Je größer und komplexer das Programm, desto länger die Zykluszeit. In diesem Fall wird eine durchschnittliche Zykluszeit von ca. 5ms erreicht. Das ist zwar für die vorliegende Aufgabe schnell genug, unterliegt aber einigen Schwankungen. Deswegen werden einige Bausteine zeitgesteuert aufgerufen. Der Zeitgesteuerte Interrupt hat eine höhere Priorität als der frei laufende Zyklus, was die Einhaltung garantiert. Die Zeit zwischen den Interrupt-Aufrufen kann in der Hardware-Konfiguration eingestellt werden, sie lässt sich aber nicht zur Laufzeit ändern. Im vorliegenden Fall beträgt die Einstellung 100ms. Eine noch kürzere Periode wäre zwar möglich, erscheint aber auf Grund der geringen Dynamik als unnötig.

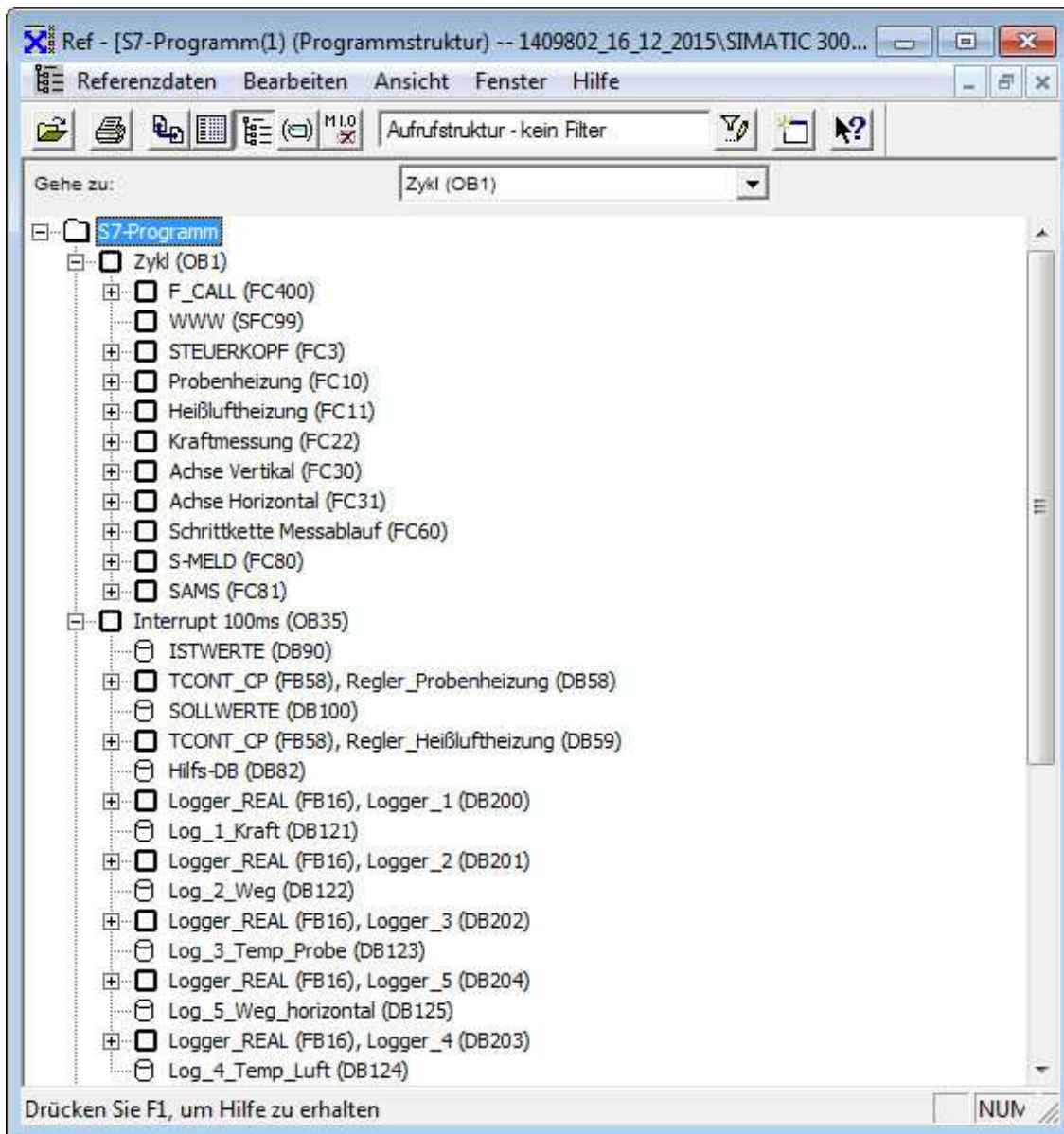


Abbildung 12 - Programmstruktur¹

Abbildung 12 zeigt die Aufrufstruktur des SPS-Programmes sowie die wichtigsten SPS-Bausteine. Zu sehen ist das Hauptprogramm „Zykl (OB1)“ mit den darin aufgerufenen Programmmodulen, sowie die Interruptroutine „Interrupt 100ms (OB35)“ mit den darin verwendeten Programmmodulen.

Bausteine im Hauptprogramm (frei durchlaufender Zyklus):

- FC400: Aufruf des Sicherheitsprogrammes.
- SFC99: Der Webserver, Bindeglied zwischen Benutzeroberfläche und SPS-Programm.
- FC3: Der Baustein namens Steuerkopf dient zur Aufnahme von allgemeinen Befehlen und von Programmsequenzen die sonst keiner Funktion zugeordnet werden können.

¹ Bildschirmfoto vom Tool „Referenzdaten Anzeigen“ im Simatic Manager

- FC10: Dient zum Ein- und Ausschalten der Probenheizung und Auswertung der Ist-Temperatur. Der Baustein zur Temperaturregelung muss für eine korrekte Funktion im Zeit-Interrupt aufgerufen werden.
- FC11: Ein- und Ausschalten der Heißluftheizung, sowie Auswertung der Ist-Temperatur. Der Regler befindet sich ebenfalls in der Interrupt-Routine.
- FC22: Hier werden Programmbausteine aufgerufen welche vom Hersteller der SPS-Eingangsmodule für DMS beigestellt wurden. Dieses Eingangsmodul hat einen erweiterten Funktionsumfang (siehe Kap 3.4). Außerdem erfolgt im FC22 auch die Weiterverarbeitung dieser Messwerte.
- FC30: Ansteuerung des Servocontrollers der Vertikalachse
- FC31: Ansteuerung des Servocontrollers der Horizontalachse
- FC60: Beinhaltet die Schrittkettensteuerung für den Messablauf
- FC80: Fehlerauswertung aller Module
- FC81: Generieren einer Sammelstörmeldung

Bausteine die im Zeit-Interrupt alle 100 ms aufgerufen werden:

- FB58: Dieser Baustein beinhaltet die PID-Regler-Algorithmen. Er wird vom SPS-Hersteller zur Verfügung gestellt. Der Regler wird in zwei Instanzen aufgerufen, mit den Instanz-DB's DB58 und DB59.
- FB16: Dieser Baustein realisiert den Datenlogger für die Aufzeichnung der Messwerte. Er wird hier fünf Mal aufgerufen, mit den Instanz-Datenbausteinen DB200 bis DB204.

3.3 Schrittkette für den Messablauf

Vor dem Start der Messung konfiguriert der Benutzer die Rahmenparameter einer Messung. Diese sind:

- Temperatur Probenheizung
- Temperatur Heißluftheizung
- Maximale Zugkraft
- Mess-Zeit
- Mess-Strecke
- Geschwindigkeit der Linearachsen
- Strecke zum Entspannen der Probe
- Abtastintervall
- Anzahl Wiederholungen
- Erhöhung der Zugkraft je Wiederholung

Danach spannt der Benutzer die Probe in die Probenaufnahme ein, schließt die Türen der Messkammer und Aktiviert die Heizung. Nun muss der Benutzer abwarten, bis die Wärme der Messkammer bis ins Innere der Probe vorgedrungen ist. Da die Proben in der Regel nicht mit Temperatursensoren präpariert werden liegt dies im Ermessen des Benutzers. Durch einen Klick auf die Starttaste auf der Benutzeroberfläche kann die Messung gestartet werden.

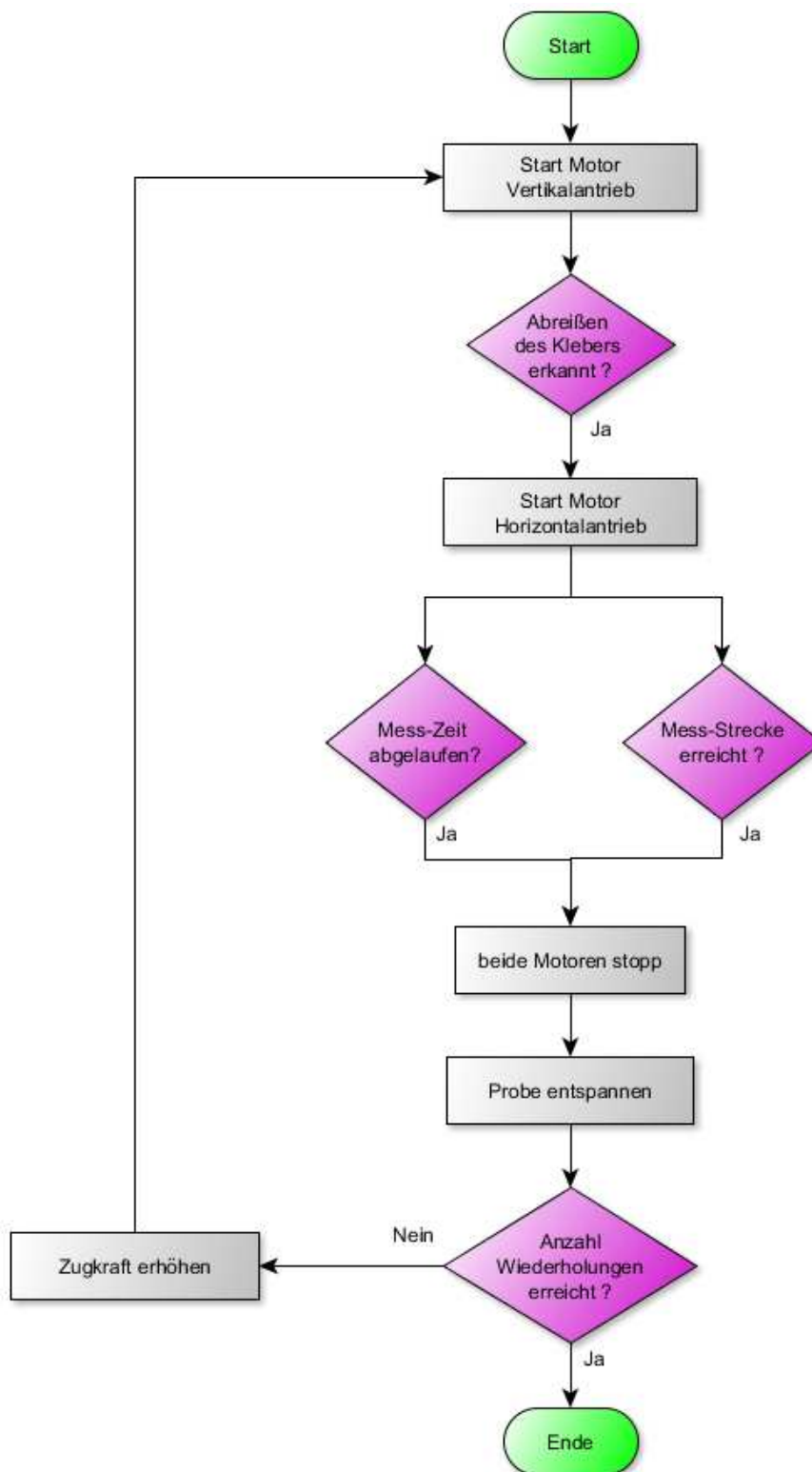


Abbildung 13 - Messablauf

Die Messung läuft folgendermaßen ab: Wie bereits erwähnt hängt die Probe Kopfüber in der Messkammer. Am rechten Ende der Probe befindet sich eine Lasche, welche in eine Zange eingespannt wurde. Die Zange ist über den Kraftsensor mit dem Vertikaltrieb verbunden. Nach Betätigung der Starttaste beginnt der Vertikaltrieb mit der eingestell-

ten Geschwindigkeit nach unten zu fahren. Dabei spannt sich die Probe. Hat sich genügend Spannung aufgebaut, dann beginnt sich die Deckschicht von der Probe zu lösen. Dieses Abreißen wird durch die SPS erkannt. Erst in dem Moment schaltet der Horizontalantrieb zu und bewegt die Probe nach rechts um den Abreißwinkel zu kompensieren. Je nachdem wie die Parameter vor der Messung eingestellt wurden wird entweder zuerst die Messzeit oder zuerst die Messstrecke abgelaufen sein. Jedes dieser beiden Ereignisse markiert das Ende einer Messung. Beide Motoren bleiben stehen. Nun fährt der Vertikaltrieb wieder ein kleines Stück nach oben, sodass die Probe wieder entspannt wird. Die Software prüft ob die Anzahl an Wiederholungen erreicht wurde. Hat der Benutzer keine Anzahl von Wiederholung eingegeben, dann endet der Ablauf hier. Die Probe kann entnommen werden. Andernfalls beginnt der Ablauf von neuem, aber mit erhöhter Zugkraft. Das Maß der Erhöhung der Zugkraft kann ebenfalls vom Benutzer eingestellt werden.

3.4 Auswertung des Kraftsensors

3.4.1 Ansteuerung der Messelektronik

Bei der Messelektronik handelt es sich um das Modul namens SIWAREX CS. Das Modul wurde speziell für den Einsatz als elektronische Waage entwickelt. Es beinhaltet bereits Funktionen wie das Trieren und das Justieren der Sensorkennlinie.

Es kann als E/A-Modul direkt an den Rückwandbus der SPS gesteckt werden. Bei Simatic S7 ist es üblich die Parameter eines E/A-Moduls im Editor „HW Konfig“ einzustellen. Beim Modul SIWAREX CS sucht man an dieser Stelle allerdings vergebens. Stattdessen müssen die Parameter durch die Anwendersoftware mittels Auftragsnummern übergeben werden. Diese Methode erscheint auf den ersten Blick nicht sehr komfortabel, bei genauerer Betrachtung aber erkennt man die Vorteile:

- Die Parameter können so auch zur Laufzeit geändert werden.
- Die Parameter können über die Benutzeroberfläche der Anwendung verändert werden.
- Eine automatische Umschaltung der Parametersätze wäre möglich (z.B. austauschbarer Sensor für verschiedene Messbereiche)

Der Datenaustausch zum Modul verläuft zum einen über die zyklische Kommunikation auf Basis der Peripherie-Adressen, zum anderen durch die azyklische Kommunikation über DP V1. Dabei verzichtet der Hersteller auf die Erklärung der Datenstrukturen aus dem E/A-Bereich. Stattdessen werden SPS-Softwarebausteine zur Verfügung gestellt, welche die Kommunikation abwickeln. Diese Bausteine müssen in das eigene Anwenderprogramm integriert werden.

Folgendes Bild soll einen Eindruck über die Komplexität der Ansteuerung des Kraftmessmoduls vermitteln:

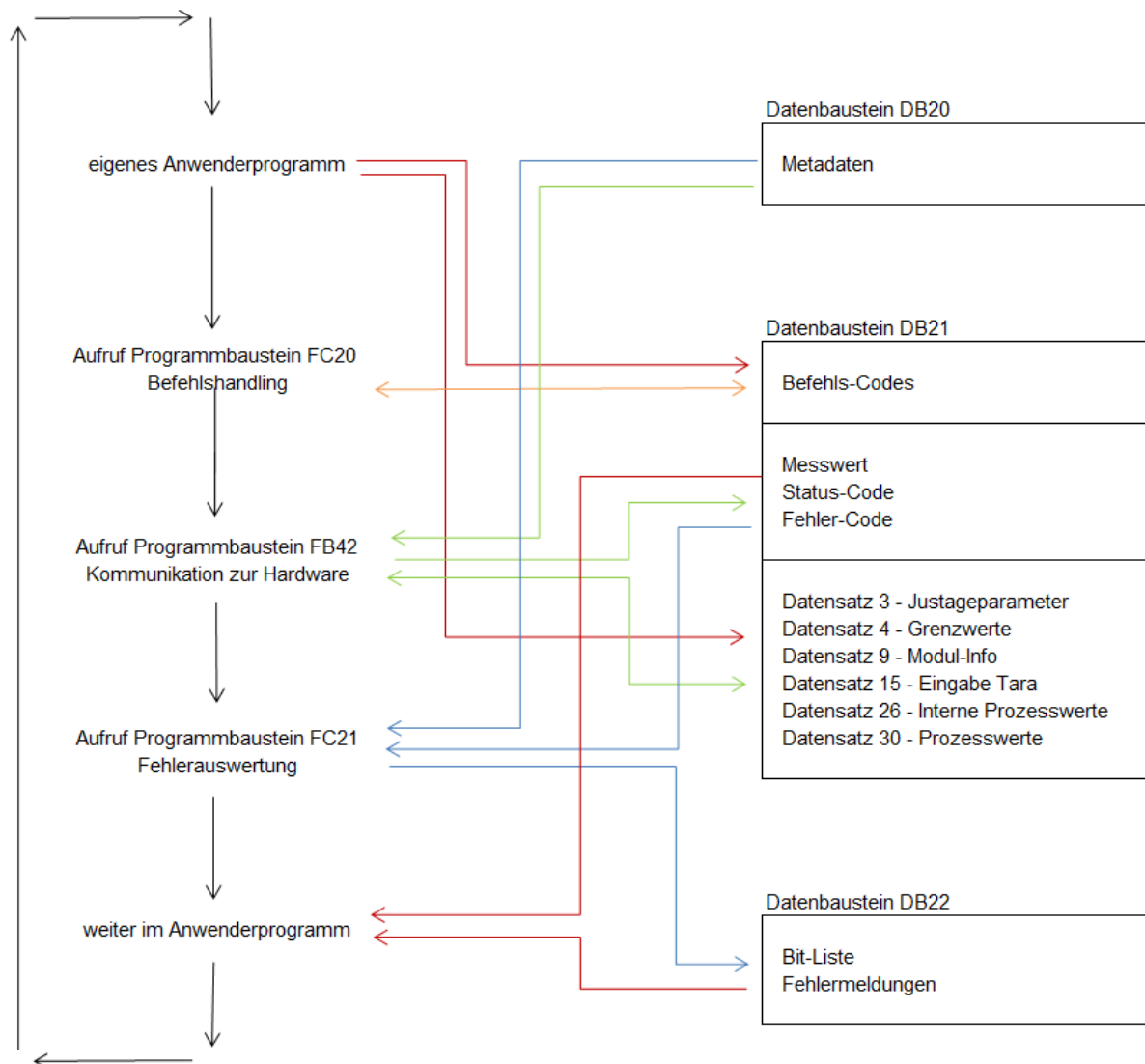


Abbildung 14 - Abhängigkeiten der Schnittstellenbausteine

Die Bausteine FC20, FC21 und FB42 werden vom Hersteller zur Verfügung gestellt. Diese werden im zyklisch im Anwenderprogramm aufgerufen. Das Anwenderprogramm füllt die jeweiligen Datensätze mit den gewünschten Parametern und schreibt die entsprechenden Befehlscodes in den DB21. Es können bis zu drei Befehle gleichzeitig eingetragen werden. Der FC20 gibt diese sequentiell an den FB42 weiter. Der FB42 führt die Befehle aus und über übernimmt die Kommunikation zur Hardware. Der FC21 wertet schließlich die Fehlercodes aus und generiert daraus eine Bit-Liste. Diese kann dann vom Anwenderprogramm weiterverarbeitet werden. Der aktuelle Messwert wird direkt an der Aufrufschnittstelle des FB42 zurückgegeben.

3.4.2 Auswertung des Kraftsensors

Wie bereits erwähnt soll der Abreißwinkel von der Maschine konstant auf 90° gehalten werden. Das ist Aufgabe des Horizontalantriebes. Allerdings kann der Horizontalantrieb nicht einfach synchron zum Vertikaltrieb bewegt werden. Bevor die Probe reißt erfährt sie erst eine gewisse Dehnung, je nach Materialbeschaffenheit und Geometrie. Das heißt der Vertikaltrieb muss erst einen gewissen Weg zurücklegen bevor die horizontale Probenachführung einsetzt. Diesen Zeitpunkt soll die Maschine automatisch erkennen, wenn möglich ohne zusätzliche Sensoren. Dazu wurde ein Algorithmus entwickelt, der den Abreißzeitpunkt vom Kurvenverlauf der Kraftmesswerte ableitet. Der Ansatz fußt auf folgender Überlegung: So lange die Probe nicht reißt und der Vertikaltrieb immer mehr Weg zurücklegt muss sich eine mechanische Zugspannung aufbauen. Die Spannung steigt immer weiter, so lange bis die Deckschicht zu reißen beginnt. Erst dann baut sich die Spannung ab bzw. steigt langsamer an. Bei klassischen Zugprüfversuchen wird dies am Abflachen der Kurve bzw. an einem Buckel erkennbar.

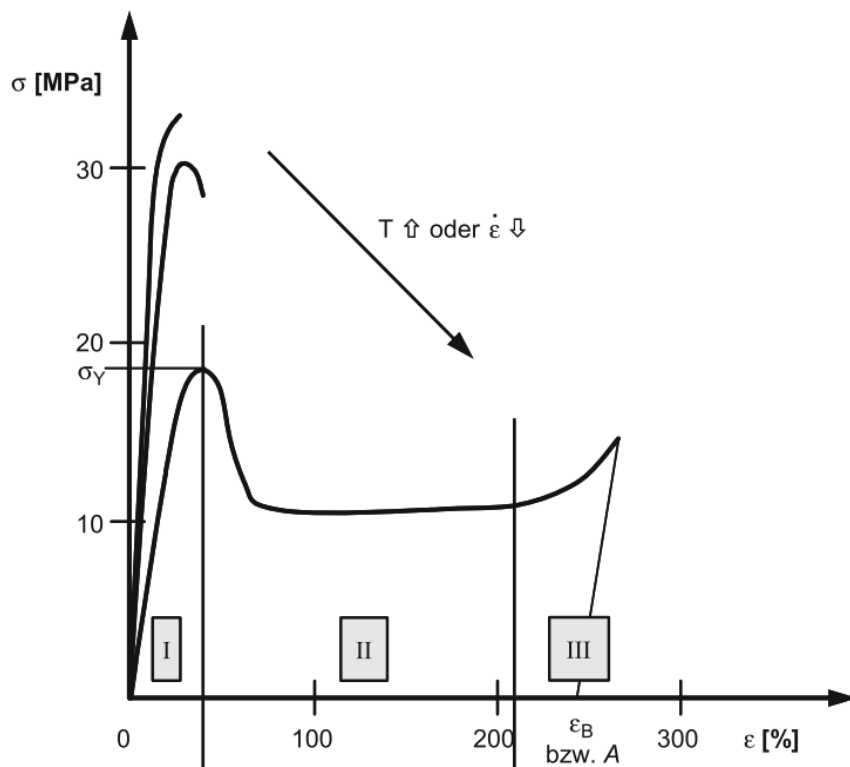


Abbildung 15 - Spannungs-Dehnungs-Diagramm [Bon2014] S.62

Abbildung 15 zeigt ein typisches Spannungs-Dehnungs-Diagramm eines Kunststoffes. Gut zu erkennen ist der steile, annähernd lineare Anstieg auf der linken Seite. Die Steilheit der Kurve richtet sich nach der Flexibilität des Probenmaterials. Auch die Abreißgeschwindigkeit hat einen Einfluss. Erst wenn das Material reißt bzw. dauerhaft verformt wird flacht die Steigung ab und bildet einen Buckel. Dieses Verhalten wird ausgenutzt um das Abreißen der Probe zu erkennen.

Die Auswertung funktioniert wie folgt:

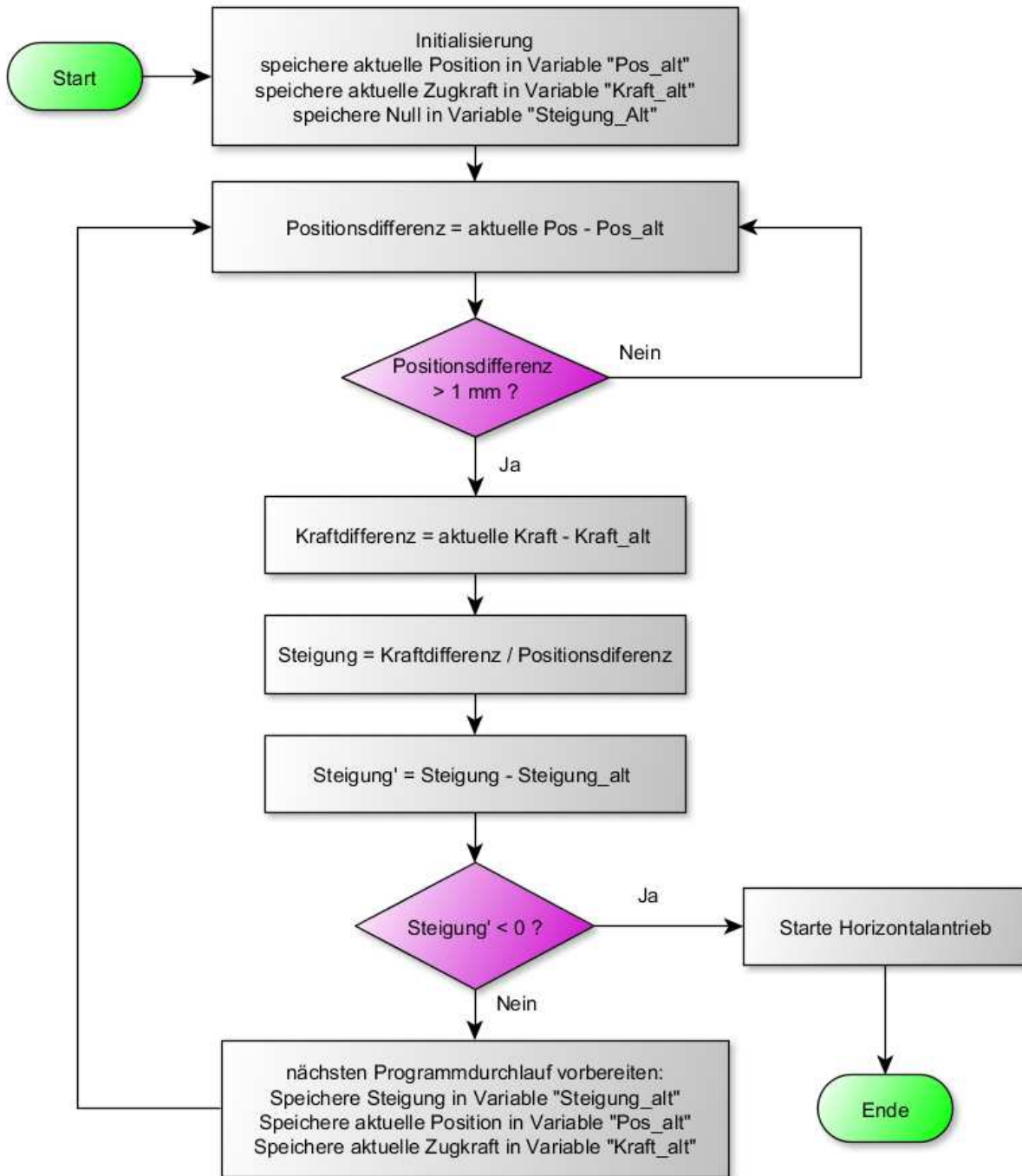


Abbildung 16 - Abreißerkennung

Zu Beginn der Messung speichert das Programm die Anfangsposition des Vertikaltriebes und die Anfangs-Zugkraft. Dann prüft es zyklisch ob sich die Position geändert hat. Wenn ja, dann wird die Distanz zur zuletzt gespeicherten Position ermittelt. Zusätzlich wird die von dieser Weg-Änderung verursachte Kraft-Änderung ermittelt, indem der gespeicherte Kraft-Wert vom aktuellen Wert abgezogen wird. Der Quotient aus Kraftänderung zu Wegänderung ergibt die aktuelle Steigung der Kurve im Kraft/Weg-Diagramm.

Der Wert der Steigung wird abgespeichert. Bei der nächsten Wegänderung wird derselbe Vorgang wiederholt. Von dieser neu gemessenen Steigung wird die Steigung aus dem letzten Durchlauf abgezogen. Der so ermittelte Wert ist die „Steigung von der Steigung“. Ist dieser Wert negativ dann wurde der Wendepunkt im Kraft/Weg-Diagramm erreicht, die Kurve wird flacher. Der Grund für diese Abflachung ist, dass das Proben-Material oder der Kleber nachgibt. Es ist also kurz vor dem Abreißen. Das ist dann der Auslöser für den Horizontalantrieb. Ab diesem Moment fährt der Horizontalantrieb bis zum Ende der jeweiligen Messung synchron mit dem Vertikaltrieb. Da während des Abreißvorganges Lufteinschlüsse und andere Unregelmäßigkeiten im Kleber zum Vorschein kommen ist eine fortlaufende Anwendung des Algorithmus über diesen Zeitpunkt hinaus nicht zielführend.

Der Algorithmus kann als Signalfussplan einfacher dargestellt werden:

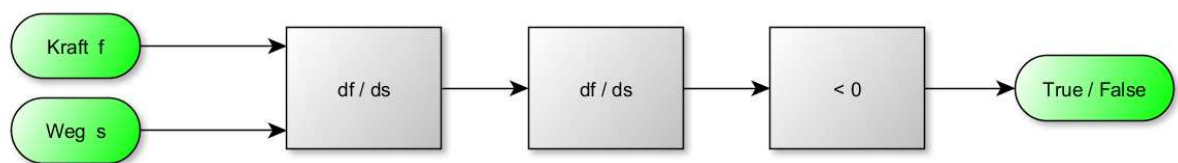


Abbildung 17 - Abreißerkennung Algorithmus

Die Praxis zeigt, dass diese Methode zu einem hinreichend genauem Einhalten des Abreißwinkels führt. Allerdings wurde festgestellt, dass sie nicht in jedem Fall funktioniert. Denn bei sehr langsamen Achsgeschwindigkeiten kommt eine Materialeigenschaft zum Tragen welche als „Kriechen“ bezeichnet wird. Dabei stellt sich eine mehr oder weniger bleibende Dehnung des Kunststoffes ein, ohne dass der Kleber reißt. In dem Fall würde der Horizontalantrieb zu früh starten.

3.5 Die Linearantriebe

3.5.1 Ansteuerung

Die Linearantriebe verfügen über einen Servomotor samt Servocontroller. Der Servocontroller selbst beinhaltet bereits eine gewisse Intelligenz und kann in bestimmten Anwendungsfällen eigenständig arbeiten. Im vorliegenden Fall jedoch erfolgt die Ansteuerung von der SPS. Für den Datenaustausch zwischen SPS und Servocontroller kommt das Protokoll Profinet zum Einsatz, da es echtzeitfähig ist und die eingesetzte SPS dieses schon in der Grundausstattung unterstützt.

Den Datenaustausch an sich übernimmt das Betriebssystem. Dazu muss der Servocontroller lediglich in der Hardwarekonfiguration der SPS eingetragen werden. Die Struktur und die Funktion der zu übertragenden Daten sind aber je nach Anwendungsfall unterschiedlich und müssen vom Anwender ausprogrammiert werden.

Dazu definiert der Hersteller das „Festo Handhabungs- und Positionier-Profil“ (FHPP). Das Protokoll beinhaltet drei Ansteuermöglichkeiten:

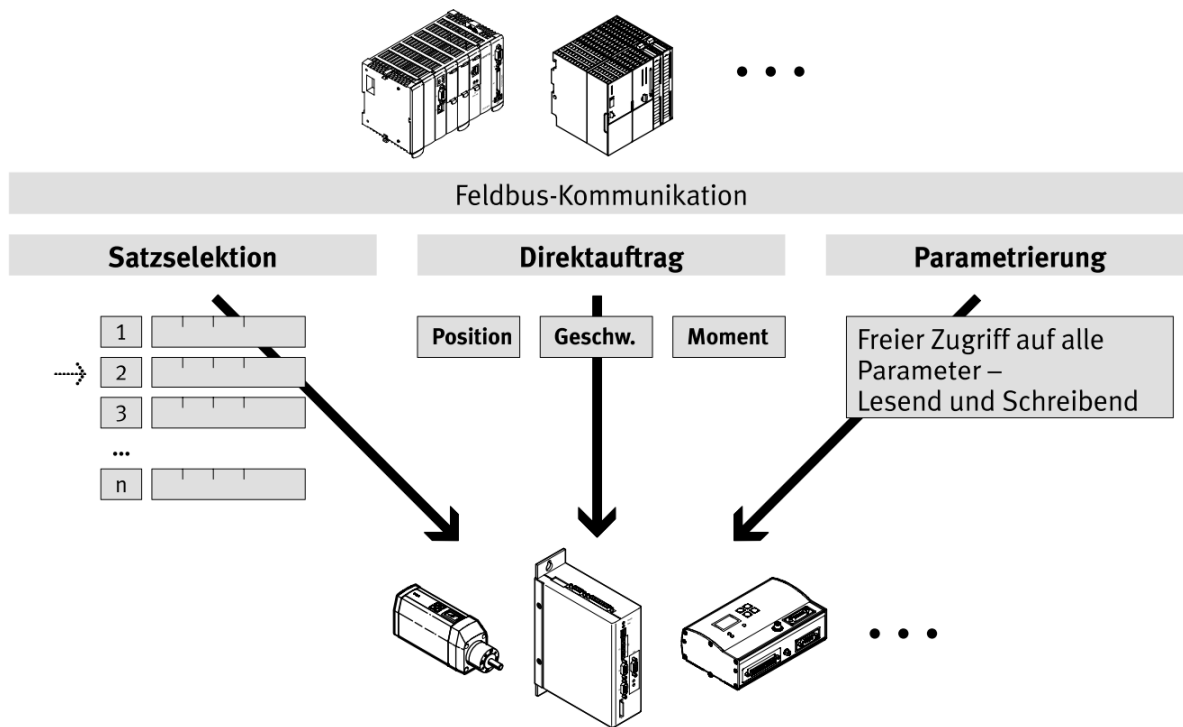


Abbildung 18 - Kommunikationsmöglichkeiten der Achsen [Fhpp2009] S. 1-3

Über die Satzselektion können Fahraufträge aus einer Tabelle ausgewählt werden. Die Tabelle muss zuvor vom Anwender erstellt und im Servocontroller abgelegt werden. Sie kann zur Laufzeit nicht bzw. nur schwer verändert werden. Deswegen wurde in diesem Fall die Ansteuerung über den Direktauftrag gewählt. Hier werden die Fahrparameter wie Zielposition, Geschwindigkeit und Drehmoment nicht aus der Satztable gelesen, sondern zyklisch von der SPS übergeben. Zusätzlich besteht die Möglichkeit über den Parameterkanal azyklisch Konfigurationsdaten im Servocontroller zu lesen und zu schreiben. Von dieser Möglichkeit wird in der vorliegenden Arbeit nicht Gebrauch gemacht. Die Parameter wurden stattdessen durch das Herstellereigene „Festo Configuration Tool“ vorgenommen.

3.5.2 Inbetriebnahme der Achsen

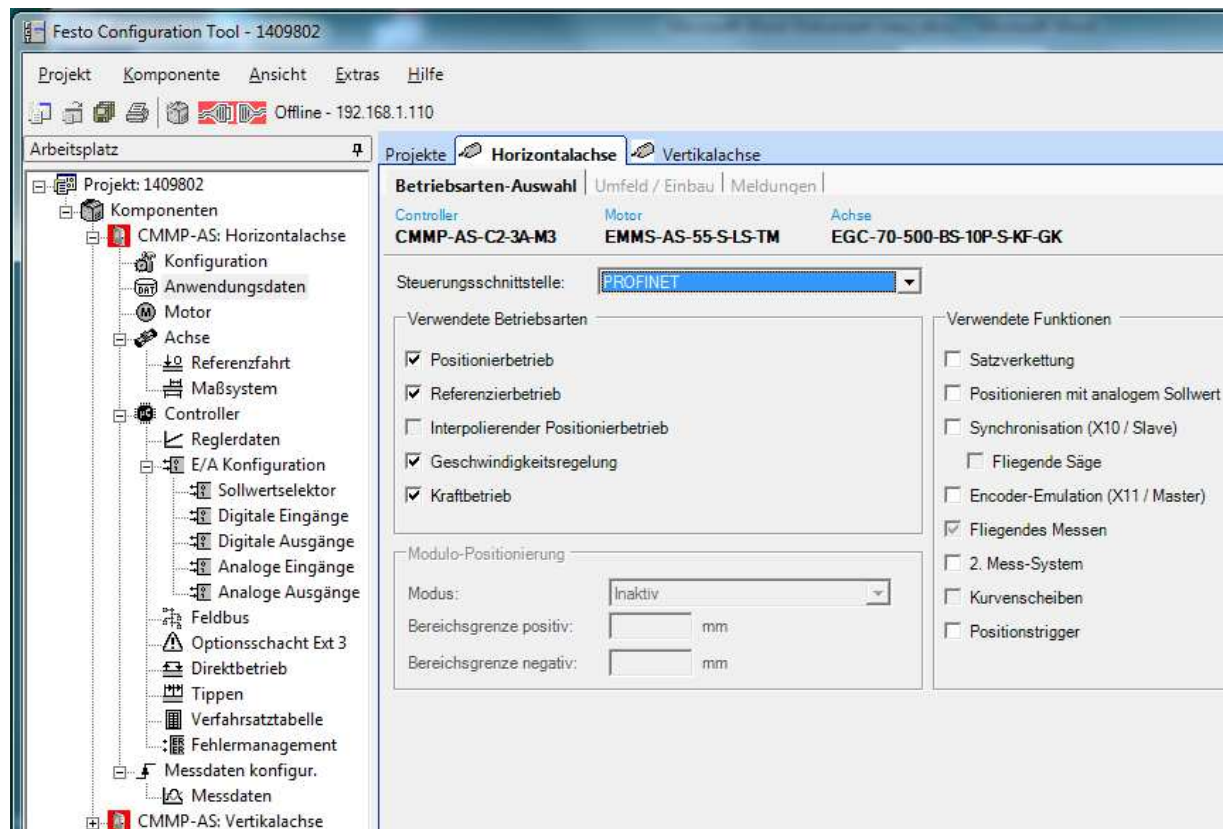


Abbildung 19 - Konfigurationstool der Achsen²

Das Bild zeigt beispielhaft die Benutzeroberfläche des Festo Configuration Tool. Auf der linken Seite befindet sich ein Navigationsbaum, welcher die Parameter in sinnvolle Bereiche aufteilt. Je nach dem welcher Bereich ausgewählt wurde erscheint auf der rechten Seite eine Maske mit den entsprechenden Einstellmöglichkeiten. Zum optimalen Einstellen der Regelparameter bietet das Tool einen Handbetrieb sowie eine Scope-Funktion. Sämtliche Parameter können als Datei gesichert werden. Die Übertragung zum Controller erfolgt per Ethernet.

3.5.3 Kraftbegrenzung

Eine Anforderung an die Zugprüfmaschine ist, dass die Zugkraft einstellbar sein muss. Man möchte damit die Probe mit einer gewissen Kraft belasten und sehen ob das Material bei länger andauernder Belastung nachgibt. Der Effekt wird auch „Kriechen“ genannt.

² Bildschirmfoto Festo Configuration Tool

Der Servocontroller bietet hierfür die Funktion „Kraftbetrieb“ [Fhpp2009] S.2-30. Dabei gibt die übergeordnete Steuerung das Drehmoment und die Drehrichtung vor. Der Abtrieb sorgt dann selbstständig für Einhaltung des Drehmomentes. Nachteilig ist jedoch die Tatsache, dass das Drehmoment nur in der Auflösung von 1%-Schritten übergeben wird. 100% entspricht dem Nenn-Drehmoment des Motors. Nachfolgende Rechnung zeigt was das für die Zugkraft bedeutet:

● **Vorschubkraft bei anliegendem Antriebsdrehmoment**

$$F_a = \frac{2\pi \cdot \eta_1 \cdot T}{Ph} \dots\dots\dots(2)$$

F_a	: Vorschubkraft	(N)
T	: Antriebsmoment	(Nmm)
Ph	: Steigung	(mm)
η_1	: Wirkungsgrad bei Rotation - Linear	

Abbildung 20 - Formel Vorschubkraft Spindelantrieb [Thk2006] S. 15-7

Der Motor hat ein Nennmoment von 0,34 Nm. Das nachgeschaltete Getriebe hat Übersetzung von 1:256. Das ergibt ein Drehmoment 87,04 Nm. Die Spindelsteigung beträgt 0,01 m/U. Die Wirkungsgrade von Kugelumlaufspindel und Getriebe sind relativ hoch und werden hier vernachlässigt. Setzt man das in obige Formel ein, dann ergibt sich eine Nenn-Vorschubkraft von 54688,85 Newton. Ein Prozent davon wären dann 546,89 Newton, was den Kraftsensor mit einem Nennbereich von +/-100 Newton schon überlasten würde. Es musste eine andere Lösung gefunden werden. Ein weiterer Versuch mittels Zweipunktregler stellte sich ebenfalls als ungeeignet heraus. Schließlich wurde folgende Methode gewählt:

Der Servocontroller wird nicht im Modus „Kraftbetrieb“ verwendet, sondern bleibt durchgehend im Modus „Positionierung“. Der Controller bekommt die Zielposition sowie die Soll-Geschwindigkeit vorgegeben. Nach dem Setzen des Start-Bits fährt der Antrieb selbstständig los in Richtung Zielposition. Während der Fahrt überprüft die SPS zyklisch den Messwert des Kraftsensors. Wenn sich der Messwert an die gewünschte Maximal-Kraft annähert dann verringert die SPS die Geschwindigkeit. Es entsteht ein Regelkreis wie er in Abbildung 21 zu sehen ist.

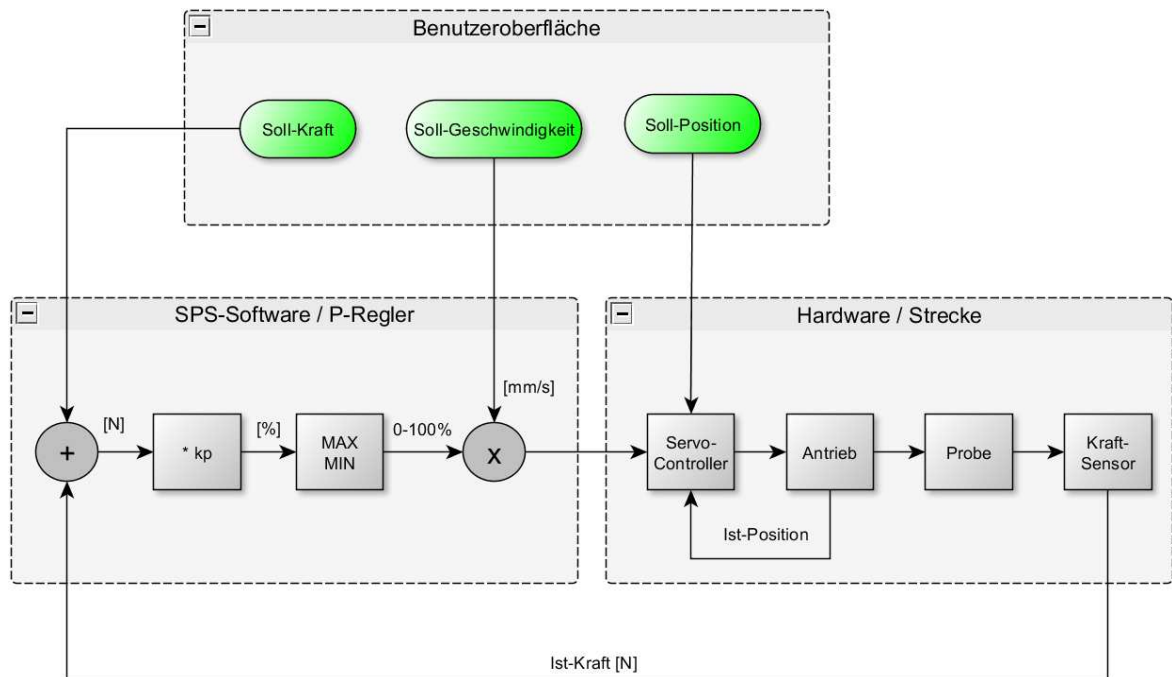


Abbildung 21 - Kraftregelung

Es handelt sich im Prinzip um einen einfachen P-Regler. Die Regelungssoftware bildet die Differenz zwischen Soll-Kraft und Ist-Kraft. Diese wird mit dem Faktor k_p multipliziert. Der Faktor ist so gewählt, dass eine Regeldifferenz von 20 N einem Stellwert von 100 % entspricht. Damit es nicht zu unerwünschten Effekten kommt wird der Wertebereich auf 0-100 % gedeckelt. Die vom Benutzer eingegebene Sollgeschwindigkeit wird dann mit dem Prozentwert multipliziert und als neuer Sollwert an den Servocontroller weitergegeben. Zusammengefasst bedeutet das: Sobald die Ist-Kraft näher als 20 Newton an die Soll-Kraft heran kommt wird die Geschwindigkeit proportional gedrosselt, gegebenenfalls bis zum Stillstand.

Es mussten allerdings weitere Probleme gelöst werden: Der Servocontroller übernimmt die Soll-Geschwindigkeit nur bei Start des Fahrauftrages. Da er hier aber Teil eines Regelkreises ist, muss die Geschwindigkeit während der Fahrt korrigiert werden. Dieser Konflikt kann relativ einfach gelöst werden indem das Start-Bit ständig invertiert wird. Der Controller erhält so laufend neue Fahraufträge. Außerdem traten Probleme mit dem Absolutwertgeber in Verbindung mit dem Getriebe auf. Durch das hohe Übersetzungsverhältnis kann die Achse nur 160 mm fahren, danach kommt der Umdrehungszähler in den Überlauf. Das Problem wurde umgangen, indem die Achse vor jeder Messung automatisch auf Null wird. Zusätzlich wurden Hardware-Endschalter montiert. Als weiterer Nachteil muss genannt werden, dass die Auflösung der Sollgeschwindigkeit in Ein-Prozent-Schritten relativ grobkörnig ist. Es sind unter Umständen Stufen im Weg-Zeit-Diagramm zu erkennen. Bei der Späteren Auswertung der Messdaten muss daher eventuell über einen gewissen Zeitraum gemittelt werden.

3.6 Probenheizung

Die Probenaufnahme besteht aus einem Aluminiumblock, in welchen eine elektrische Heizung integriert wurde. Die Heizung wird durch ein Solid State Relais angesteuert. Mittels PT100 soll die Temperatur geregelt werden. Da die Heizleistung sehr großzügig bemessen ist würde ein einfacher Zweipunktregler zu extremen Überschwingen führen. Deswegen fiel die Wahl auf einen quasi-kontinuierlichen PID-Regler. Der SPS-Hersteller liefert hierzu bereits den passenden Softwarebaustein mit. Es handelt sich um den FB58 namens „TCONT_CP“. Ein Blick in das zugehörige Handbuch [SiePid2003] verrät, dass der Baustein in regelmäßigen Abständen bearbeitet werden muss. Er wird daher im Zeitinterrupt alle 100 Millisekunden aufgerufen. Der Baustein bringt auch gleich einen PWM-Generator für die Ansteuerung des Solid State Relais mit.

Die integrierte Selbstoptimierung [SiePid2003] S. 3-1 erleichtert das Einstellen der Regelparameter ungemein. Während der Selbstoptimierung muss ein Aufheizvorgang gestartet werden. Der FB58 nimmt die Sprungantwort der Strecke auf. Anhand der Lage des Wendepunktes werden die Regelparameter ermittelt. Ein in Step7 integrierter Wizard stellt eine grafische Benutzeroberfläche zur Selbstoptimierung. Die Optimierungsfunktion kann aber auch unabhängig davon zur Laufzeit verwendet werden.

3.7 Heißluftheizung

Das Heizgebläse selbst besitzt bereits einen integrierten Controller mit Temperaturregelung. Die SPS übergibt dem Controller die Solltemperatur sowie die Luftmenge über zwei Analogsignale im Bereich von 4 bis 20mA.

Es stellte sich heraus, dass der eingebaute Temperaturregler nicht für diesen Einsatzzweck geeignet ist. Er ist für Temperaturen von 50 bis 650 °C ausgelegt. In der Zugprüfmaschine sollen aber Temperaturen von 30 bis 80°C herrschen. Das stellt für den Regler den untersten Grenzbereich dar, sodass die Analogsignale nicht ausreichend genau interpretiert werden.

Als Abhilfe wurde beschlossen das Gerät vom Betriebsmodus Regelung (closed loop) in die Betriebsart Steuerung (open loop) umzustellen. Auf diese Art wird der Analogwert nicht mehr als Solltemperatur sondern als Stellwert für die Heizleistung interpretiert. Die Regelung erfolgt nun durch die SPS. Auch hier kommt wieder der bewährte PID-Regelbaustein FB58 zum Einsatz.

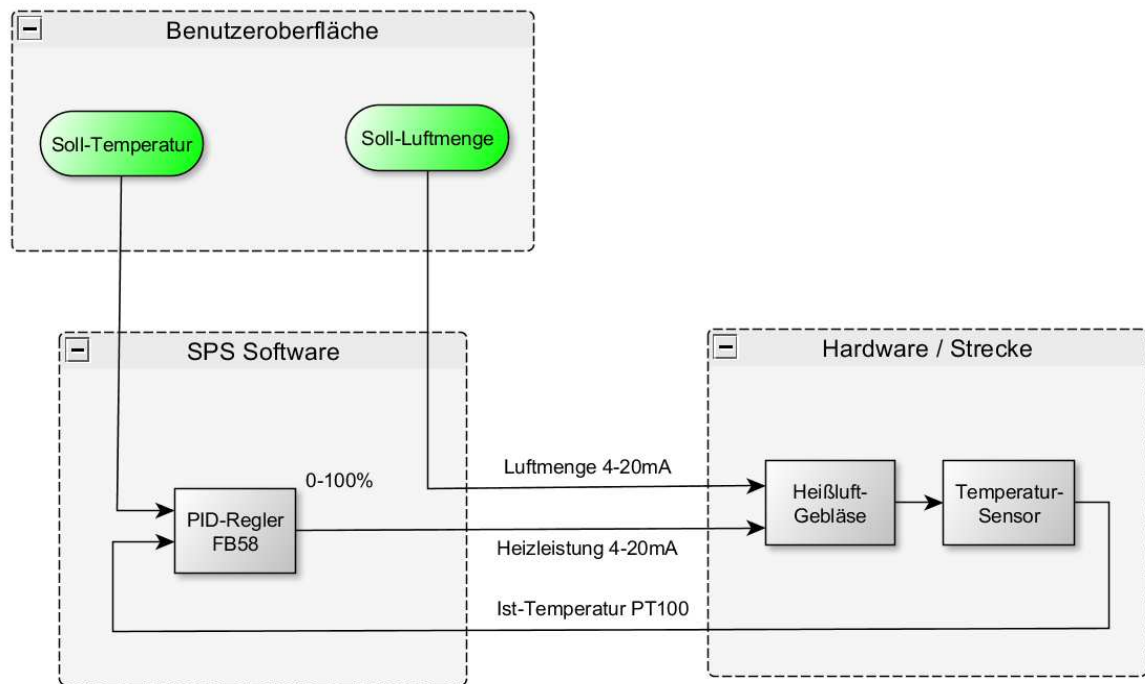


Abbildung 22 - Regelung Heißlufttheizung

Auf diese Art lässt sich die Regelung auf den gewünschten Bereich anpassen, die Stabilität wird verbessert.

3.8 Aufzeichnung der Messwerte

Die Kernfunktion der Zugprüfmaschine ist natürlich die Aufnahme der Messwerte. Im Lastenheft werden zwei Betriebsarten gefordert:

1. Konstante Abreißgeschwindigkeit → Beobachtung der auftretenden Kräfte
2. Konstante Zugkraft → Beobachtung der sich einstellenden Abreißgeschwindigkeit

Es werden folgende Größen aufgezeichnet:

- Zugkraft
- Position Horizontalachse
- Position Vertikalachse
- Temperatur Probenheizung
- Temperatur Heißlufttheizung

Nun stellt sich die Frage nach den anfallenden Datenmengen. Diese sind von der Betriebsart abhängig. In der ersten Betriebsart können je nach eingestellter Abreißgeschwindigkeit sehr dynamische Wertänderungen auftreten. Die Messung dauert jedoch in der Regel nur einige Sekunden. In der zweiten Betriebsart geht es um das bereits erwähnte Kriechverhalten von Kunststoffen. Dabei wird die Probe mit einer bestimmten Kraft beaufschlagt und teilweise mehrere Stunden auf Spannung gehalten. Dafür ist in dieser Betriebsart eine geringere Dynamik in den Messwerten zu erwarten.

Die erstere Betriebsart verlangt ein kurzes Messintervall, die zweite Betriebsart eine lange Aufzeichnungsdauer. Beide Anforderungen gemeinsam würden schnell zu unhandlich großen Datenmengen führen. Es musste der richtige Kompromiss gefunden werden. Folgende Methode erwies sich als praxistauglich:

- Es werden von jeder Messgröße maximal 2000 Werte aufgenommen. Das begrenzt den notwendigen Speicherbedarf.
- Die Messwerte werden in einem Grund-Intervall von 100 Millisekunden aufgenommen. Über die Benutzeroberfläche kann das Intervall auf ein vielfaches verlängert werden. So ist auch eine Messung über längere Zeit möglich.

Für diese Aufgabe wurde ein Datenlogger-Baustein entwickelt. Für jede aufzuzeichnende Messgröße ist der Baustein einmal aufzurufen. Bei jedem Aufruf sind ein eigener Instanz-Datenbaustein sowie ein Speicherbereich für die Ablage der Messwerte anzugeben. Abbildung 23 zeigt die Aufrufchnittstelle am Beispiel der Kraftmessung.

Netzwerk 5 : Kraft aufzeichnen

CALL FB	16 , DB200	Logger_REAL / Logger_1
Istwert	:=DB90.DBDO	"ISTWERTE".Kraft
Datenbaustein	:=DB121	Log_1_Kraft
Datenwerte	:= "Log_1_Kraft".Werte	"Log_1_Kraft".Werte
Enable	:=M100.2	SK1/03
MAX_WERT	:=DB121.DBD8000	"Log_1_Kraft".Max_Wert
MAX_INDEX	:=DB121.DBW8004	"Log_1_Kraft".Max_Index

Abbildung 23 - Schnittstelle Datenlogger

Der Befehl „CALL FB16, DB200“ ruft den Baustein FB16 mit dem Instanz-Datenbaustein Nr. 200 auf. Darunter folgen die Übergabeparameter an den Funktionsbaustein.

Am Parameter „Istwert“ wird die aufzuzeichnende Messgröße im Format REAL übergeben. Am Eingang „Datenbaustein“ muss derjenige Datenbaustein angegeben werden, in dem die aufgezeichneten Messwerte abgelegt werden. Hier ist das der DB121. Der Parameter „Datenwerte“ definiert den Bereich innerhalb des Datenbausteines, der für die Messwerte zur Verfügung steht. Als Datentyp wird ein Pointer vom Typ ANY erwartet. Ein Any-Pointer beinhaltet nicht nur die Adresse sondern auch die Länge eines Datenbereiches. Mit dem Bit „Enable“ wird die Aufzeichnung gestartet. Die Parameter „MAX_WERT“ und „MAX_INDEX“ werden vom Funktionsbaustein an das Aufrufende Programm zurückgegeben. Sie beinhalten den größten Messwert bzw. die Anzahl der aufgezeichneten Messwerte. Diese Parameter werden von der Benutzeroberfläche verwendet um die Achsen des X/Y-Diagrammes zu skalieren.

4 Visualisierung

4.1 Auswahl der Visualisierungs-Plattform

Als Benutzeroberfläche sollte ein Beliebiger Rechner dienen, vorzugsweise die im Unternehmen vorhandenen Laptops. Es wurden folgende Möglichkeiten in Betracht gezogen:

- Eine der in der Industrie üblichen HMI/SCADA-Softwarepakete
- Benutzeroberfläche geschrieben in einer Hochsprache
- Benutzeroberfläche erstellt mittels Framework
- Verwendung von Webseiten als Benutzeroberfläche

Kommerzielle HMI/SCADA-Softwarepakete sind zwar auch auf einem Laptop lauffähig, allerdings braucht man zur Installation sowohl eine Lizenz als auch Administratorrechte. Dies kann zu einem Problem werden wenn verschiedene Personen abwechselnd mit verschiedenen Rechnern arbeiten möchten. In dem Fall müssten auf allen betreffenden Rechnern die Software und die Lizenzen installiert werden bzw. die Lizenzen ständig weitergereicht werden. Das kann für den Administrator eines Unternehmens sehr arbeitsintensiv werden. Alternativ könnte die Benutzeroberfläche natürlich auch in einer der vielen Programmiersprachen, wie z.B. C oder VB, geschrieben werden. Hier steht der Programmierer vor der Herausforderung die Kompatibilität zu den unterschiedlichen im Umlauf befindlichen Systemen zu gewährleisten. Je nach Hardware und Betriebssystem müssen unterschiedliche Compiler oder Bibliotheken verwendet werden. Dieses Problem lässt sich durch den Einsatz von Laufzeitumgebungen wie z.B. Java oder .NET umgehen. Darüber hinaus sind einige davon kostenfrei nutzbar. Aber sowohl die Laufzeitumgebung als auch der Quellcode müssen auf die betreffenden Rechner verteilt und aktuell gehalten werden. Auch dieser administrative Aufwand ist obsolet wenn man auf Web-Technologien zurückgreift. Da die hier verwendete SPS bereits einen Webserver mitbringt liegt der Gedanke nahe. In dem Fall übernimmt der Webbrowser die Funktion einer Laufzeitumgebung. Die Benutzeroberfläche ist damit eine Webseite. Das hat der Vorteil, dass die Quellcodes (die Webseite) nicht vorher auf die einzelnen Arbeitsrechner verteilt werden muss. Stattdessen holt sich der Browser selbstständig immer die aktuelle Version der Benutzeroberfläche. Ein positiver Nebeneffekt ist, dass die Benutzeroberfläche mit dieser Methode auch auf Tablets und Smartphones lauffähig ist. Aus oben genannten Gründen wurde die Web-Technologie als Visualisierungswerkzeug gewählt. Es folgt eine Zusammenfassung der Vor- und Nachteile in Tabellenform.

Methode	Lizenzkosten	Program- mieraufwand	Bemerkung
HMI/SCADA	Hoch	Gering	Abhängigkeit vom Hersteller
Compiler-Sprache	Gering	Hoch	Unabhängig aber aufwendig
Laufzeitumgebung	Gering	Mittel	Vom Framework abhängig
Web-Technologie	Gering	Mittel	Offener Standard

Tabelle 1 - Vergleich der Visualisierungsmethoden

Methode	Pro und Contra
HMI/SCADA	<ul style="list-style-type: none"> + hoher Vorfertigungsgrad - hohe Lizenzkosten - Beschränkung auf Funktionsumfang des Herstellers - bei Einsatz verschiedener Rechner mehrfache Installation nötig - dadurch umständliche Handhabung der Programme und Lizenzen - Benutzerrechte zur Installation nötig - Im Falle einer Änderung muss die neue Version verteilt werden
Compiler-Sprache	<ul style="list-style-type: none"> - Niedriger vorfertigungsgrad, dadurch hoher Programmieraufwand + keine Beschränkung im möglichen Funktionsumfang + keine Lizenzkosten (bei GPL-Compilern) - bei Einsatz verschiedener Rechner mehrfache Installation nötig, - dadurch umständliche Handhabung der Programme - Benutzerrechte zur Installation nötig - Im Falle einer Änderung muss die neue Version verteilt werden
Laufzeitumgebung	<ul style="list-style-type: none"> +/- mittlerer Vorfertigungsgrad - Beschränkung auf Funktionsumfang des Herstellers + keine Lizenzkosten (bei GPL-Frameworks) - bei Einsatz verschiedener Rechner mehrfache Installation des Frameworks nötig - dadurch umständliche Handhabung der Programme - Benutzerrechte zur Installation des Frameworks nötig - Im Falle einer Änderung muss die neue Version verteilt werden
Web-Technologie	<ul style="list-style-type: none"> +/- mittlerer Vorfertigungsgrad + keine Lizenzkosten - Beschränkung auf Funktionsumfang des Browsers + Web-Browser mittlerweile auf jedem System vorhanden + beliebige Rechner, auch Tablets und Smartphones verwendbar + keine Installation notwendig + keine Benutzerrechte notwendig + Im Falle einer Änderung keine Verteilung/Neuinstallation nötig

Tabelle 2 - Pro und Contra der Visualisierungsmethoden

4.2 Grundlagen Web-Programmierung

Dieses Kapitel soll einen kurzen Einblick in die Web-Programmierung geben. Für detaillierte Informationen wird auf die zahlreich vorhandenen Nachschlagewerke [Web2015] [Sel2015] verwiesen.

Moderne Webseiten sind ein Mix aus verschiedenen Code-Sprachen. Die drei häufigsten sind:

- HTML (Hyper Text Markup Language)
- CSS (Cascading Style Sheets)
- JavaScript

4.2.1 HTML

Über die Textauszeichnungssprache HTML wird in der Regel das Gerüst der Webseite definiert. HTML beschreibt die Strukturierung und den Text-Inhalt einer Webseite. Die einzelnen Strukturelemente werden durch sogenannte Tags repräsentiert. Informationen können in Tags eingerahmt werden. z.B:

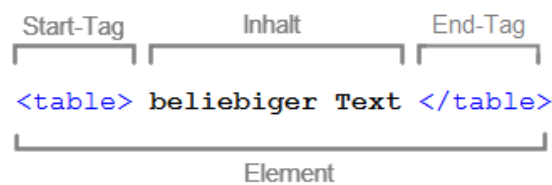


Abbildung 24 - Beispiel eines einzelnen HTML-Elementes

Elemente können beliebig verschachtelt werden um größere Strukturen zu formen.

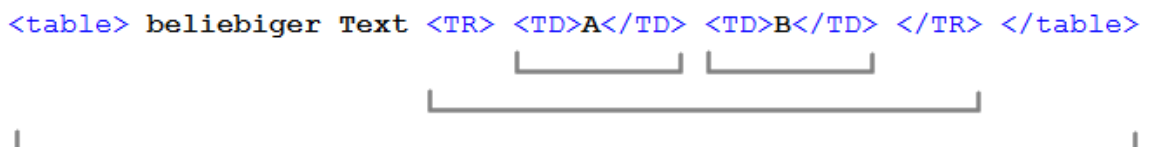



Abbildung 25 - Beispiel Verschachtelte HTML-Elemente

Die Tag-Namen sind nicht beliebig sondern sind genormt [W3C]. Der Tag-Name kann als eine Art Datentyp oder Objekt-Typ betrachtet werden. Durch die Verschachtelung können Informationen gruppiert werden. Aus der so geformten Daten-Struktur erzeugt der Browser zur Laufzeit entsprechend verschachtelte Objekte welche schlussendlich zum Rendern der Webseite dienen. Skript-Sprachen können über die standardisierte DOM-Schnittstelle zur Laufzeit auf diese Objekte einwirken.

4.2.2 CSS

Mit CSS-Anweisungen lassen sich bestimmte Eigenschaften von HTML-Elementen genauer spezifizieren. Dabei bedient es hauptsächlich die grafische Darstellung der Elemente. CSS-Befehle können entweder direkt an den jeweiligen HTML-Elementen angegeben werden, oder sie werden an einer zentralen Stelle gesammelt. Letzteres ist zu bevorzugen weil durch Synergien Speicherplatz gespart und die Programmpflege erleichtert werden kann. Folgender Beispiel-Code erzeugt ein Paragraph-Element, dessen Text blau dargestellt wird:

```
<p style="color:blue;" > beliebiger Text </p>
```



CSS-Anweisung

Abbildung 26 - Beispiel CSS-Anweisung

4.2.3 JavaScript

JavaScript ist eine Programmiersprache, welche von den zurzeit gängigsten Webbrowsern unterstützt wird. Die Sprache wurde ursprünglich für den Einsatz in Webseiten entwickelt. Seine Befehle sind aber universell gestaltet, sodass sie nicht nur auf Webseiten beschränkt sondern auch in anderen Bereichen der IT angewendet wird. Mittels JavaScript können die von HTML erzeugten Elemente (Objekte) zur Laufzeit beeinflusst werden. Mit Skriptsprachen ergeben sich unzählige Möglichkeiten. Hier einige Beispiele:

- Verzweigungen eines Auswahl-Menüs ein- und ausblenden
- Inhalt von Textfeldern verändern / aktualisieren
- Es können große Datenstrukturen im Hintergrund der Webseite dynamisch verwaltet werden, wie z.B. Messwertpuffer für Kurvenanzeige.
- Es können HTML-Elemente zur Laufzeit gelöscht oder hinzugefügt werden
- Es können Ereignisse des Browsers und Benutzereingaben abgefragt werden, wie z.B. Laden und Schließen der Webseite, Timer-Events, Mausbewegung, Klicks, Tastatureingaben usw.
- Kommunikationsmöglichkeit mit anderen Servern (AJAX, Websocket)

Einige grundlegende Eigenschaften von JavaScript [Ack2015]:

- Dynamische Typisierung
- Objektorientiert aber ohne native Klassen (Klassen können emuliert werden)
- Basiert auf Prototypen
Neue Objekte erzeugt man, indem man von bestehenden Objekten ableitet und diesen neue Funktionen hinzufügt.
- Sowohl imperative als auch funktionale Programmierung möglich

Der Skript-Code kann direkt im HTML-Dokument platziert werden. Dazu müssen die Befehle durch das <script>-Element umschlossen werden. Aber wie bei CSS sollte der

Script-Code zentral platziert werden. Er kann auch in eine eigene Datei ausgelagert werden.

```
<script>
  var x=5;
  var y=6;
  var z=x+y;
  alert(z);
</script>
```

Abbildung 27 - Beispiel zur JavaScript-Syntax

4.2.4 Webseite

Eine Webseite wird durch das Zusammenspiel aller dieser Sprachen erzeugt. Moderne Webseiten beinhalten oft so viel Skript-Code, dass man auch von einer Web-Applikation spricht. Die Anweisungen der Webseite werden in menschenlesbarem Format als Text abgelegt. Sie müssen nicht in Maschinencode Übersetzt werden sondern werden zur Laufzeit vom Browser interpretiert. Die Texte-Informationen werden zusammen mit den in der Webseite verwendeten Mediendateien (z.B. Bilder) am Webserver hinterlegt. Wird die Webseite dann von einem beliebigen Rechner abgerufen, dann werden die Quellcodes vom Webserver zum Benutzerrechner übertragen. Erst am Client-Rechner wird die Webseite gerendert und das Skript-Programm ausgeführt. Der Webserver selbst verbraucht also nicht unbedingt viel Rechenleistung.

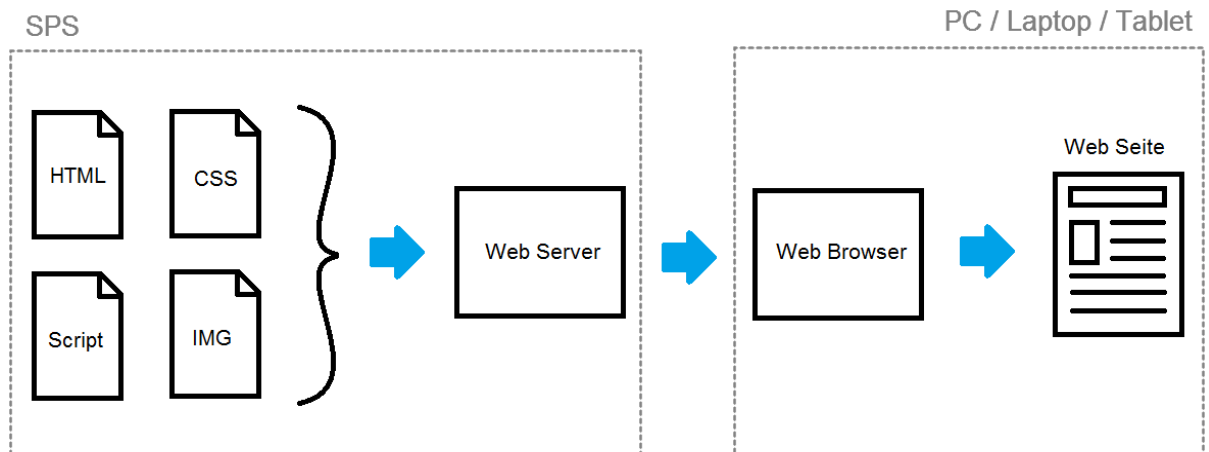


Abbildung 28 - Webserver und Webbrowser

Bei der Zugprüfmaschine läuft der Webserver als Teil des herstellereigenen Betriebssystems auf der SPS. Der Quellcode der Webseite wird innerhalb der üblichen Step7-Datenbausteine abgelegt. Der Zugriff erfolgt über die Ethernet-Schnittstelle der SPS. Ein handelsüblicher Access Point erlaubt den Zugriff über WLAN.

4.3 Das Tool Web2PLC

Das Tool S7-Web2PLC ist in der Entwicklungsumgebung von Step7 enthalten. Es erfüllt folgende Aufgaben:

- Import von Symbol-Informationen aus dem S7-Projekt für den symbolischen Zugriff von der Webseite auf die SPS-Variablen.
- Anlegen von ENUM-Typen (Aufzählungs-Variablen bzw. Textlisten)
- Interpretieren von AWP-Kommandos
- Unterstützung der Sprachumschaltung
- Einlesen der Quell-Dateien der Webseite und umwandeln in Datenbausteine
- Erzeugung von Metainformationen und Ablage im Datenbaustein
- Laden der Datenbausteine in den Speicher der SPS

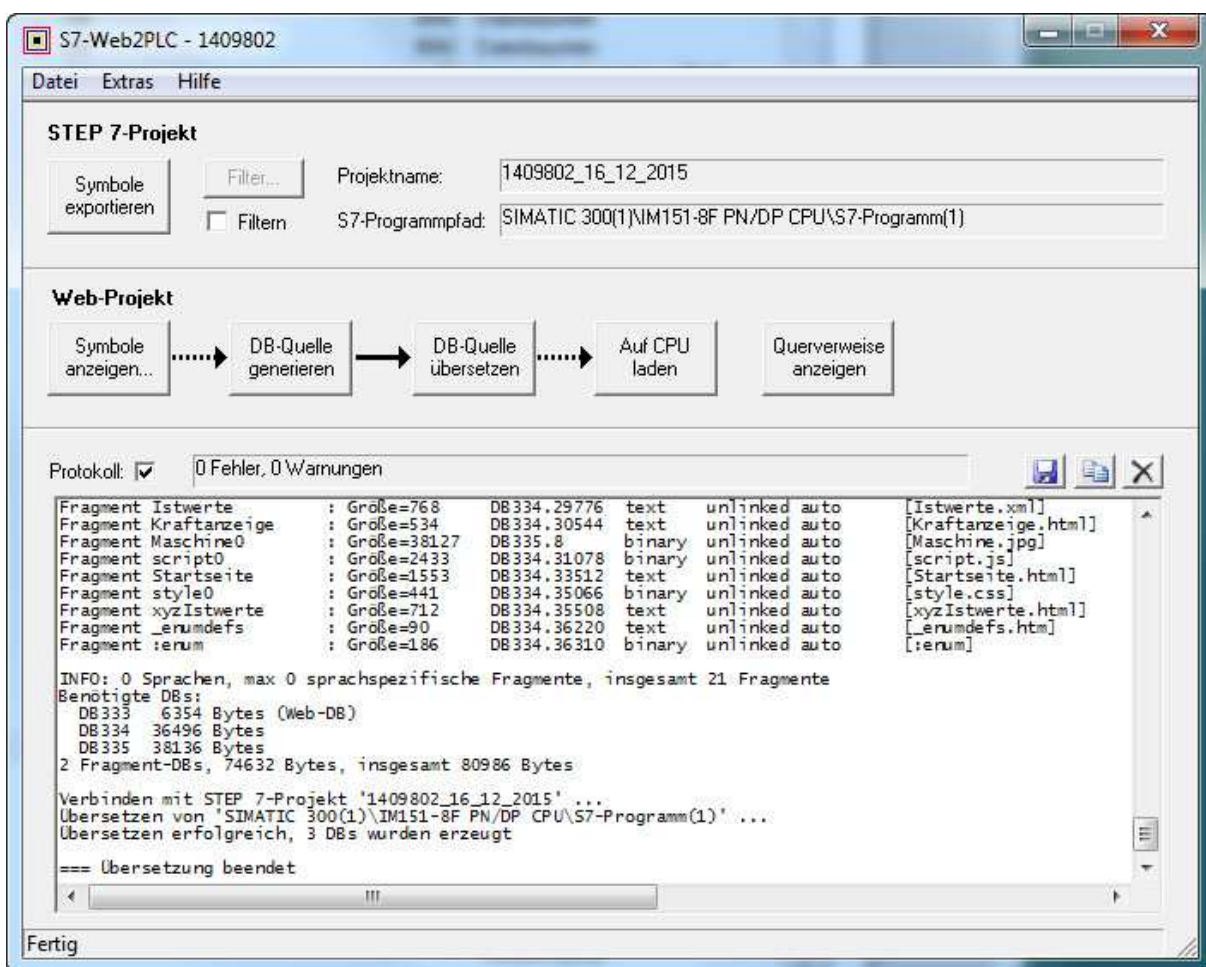


Abbildung 29 - S7-Web2PLC³

³ Bildschirmfoto Step7 Web2PLC

4.3.1 Arbeitsschritte

1. Programmieren einer Webapplikation, evtl. unter Zuhilfenahme von geeigneten Tools
2. Einfügen von AWP Kommandos für dynamische Seiten.
3. Generieren der DBs (Web-Control-DB und Fragment-DBs) mittels Web2PLC. Standardmäßig werden die DBs 333 und 334 erzeugt. Die DBs enthalten Metainformation sowie die komplette Webapplikation inklusive der Bilder und den dynamischen und statischen Teilen der Webapplikation.
4. Die Funktion SFC 99 im Anwenderprogramm aufrufen. Der SFC 99 initialisiert den Webserver der CPU.
5. Laden der Bausteine in die CPU.
6. Webseite im Browser aufrufen. Der Aufruf erfolgt über die Eingabe der IP-Adresse der CPU.

4.3.2 Einstellungen in Web2PLC

Bevor die Arbeit mit Web2PLC losgehen kann muss erst ein Projekt angelegt werden. Standardmäßig werden die Dateien des Web2PLC-Projektes innerhalb des zugehörigen S7-Projektes abgelegt. Dadurch werden die Daten beim Archivieren gleich mit gesichert.

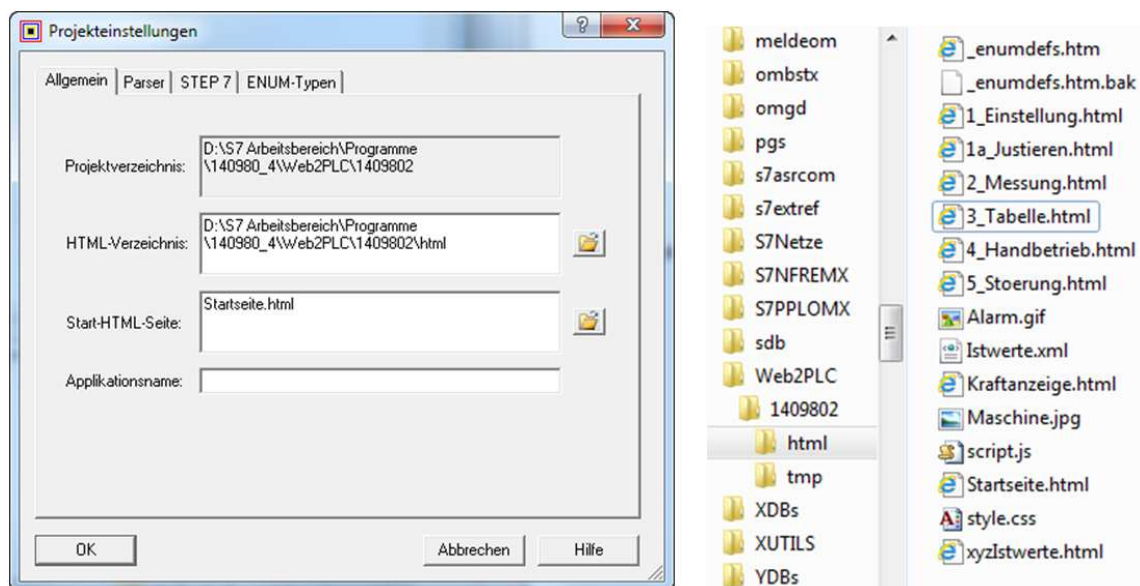


Abbildung 30 - Verzeichnisse

In dieses Verzeichnis werden alle Ressourcen des Web-Projektes abgelegt. Die Aufgabe kann natürlich in mehrere HTML-Dateien aufgeteilt werden. In den Projekteinstellungen erfolgt der Verweis auf die Start-Seite.

4.3.3 AWP-Kommandos

Die Webseiten sollen als Benutzeroberfläche dienen. Dazu müssen die Benutzereingaben von der Webseite in den Speicher der SPS gelangen und umgekehrt. Einen solchen Mechanismus implementiert Siemens in Form von AWP-Kommandos. Diese Kommandos werden üblicherweise im HTML-Code verwendet, können aber auch in beliebigen anderen Ressourcen, wie z.B. in JavaScript oder in Textdateien, verwendet werden.

4.3.3.1 SPS-Variable lesen

Im einfachsten Fall wird ein AWP-Kommando als Platzhalter für einen Wert einer Variablen verwendet. Wenn die Webseite (oder eine beliebige andere Text-Ressource) abgerufen wird, dann ersetzt der Webserver den Platzhalter durch den Inhalt der entsprechenden SPS-Variablen.

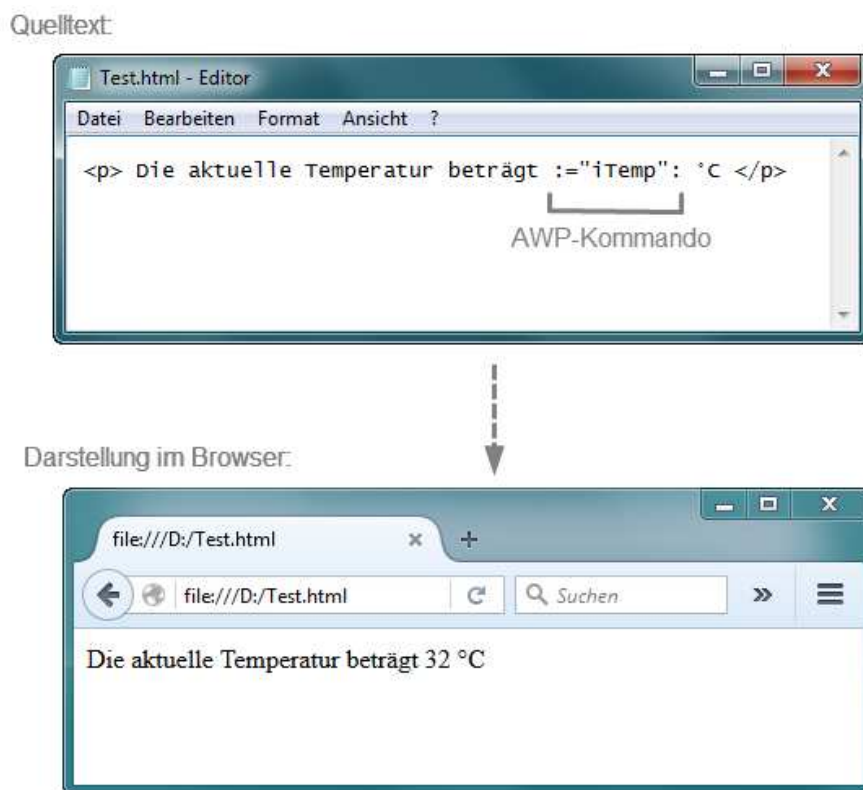


Abbildung 31 - Beispiel SPS-Variable lesen

Die im Beispiel verwendete Variable namens „iTemp“ muss im S7-Projekt vorhanden sein, d.h. es muss einen entsprechenden Eintrag in der Symboltabelle oder in einem Datenbaustein geben. Das Tool S7-Web2PLC liest die Symbolinformation aus dem S7-Projekt und kennt damit die Adresse der Variablen. Die Informationen werden im „WEB_CONTROL_DB“ hinterlegt, sodass der Webserver die richtigen Werte zur Laufzeit einsetzen kann. Der Webserver übernimmt dabei die Konvertierung vom Binärformat in Textformat. Es können SPS-Variablen mit beliebigen Datentypen verwendet werden, z.B. INT, REAL, und auch STRING.

4.3.3.2 Prozesswerte zyklisch aktualisieren

Bei der oben beschriebenen Vorgehensweise wird der aktuelle Wert nur bei Anforderung der Webseite eingetragen. Liegt die Seite einmal im Speicher des Browsers bleibt sie statisch. In der Regel möchte der Benutzer aber immer möglichst aktuelle Werte sehen. Dazu könnte man z.B. die Webseite in kurzen Abständen ständig neu anfordern. Dies würde aber zu einem Unangenehmen Flackern der Anzeige führen und Benutzeraktionen wie Scrollen wären unmöglich. Stattdessen greift man auf JavaScript zurück. Mit Hilfe des Objektes „XMLHttpRequest“ kann eine Web-Ressource im Hintergrund angefordert werden. Deren Informationen werden dann durch DOM-Manipulation in die aktuell angezeigte Webseite eingefügt, ohne dass diese neu geladen werden muss.

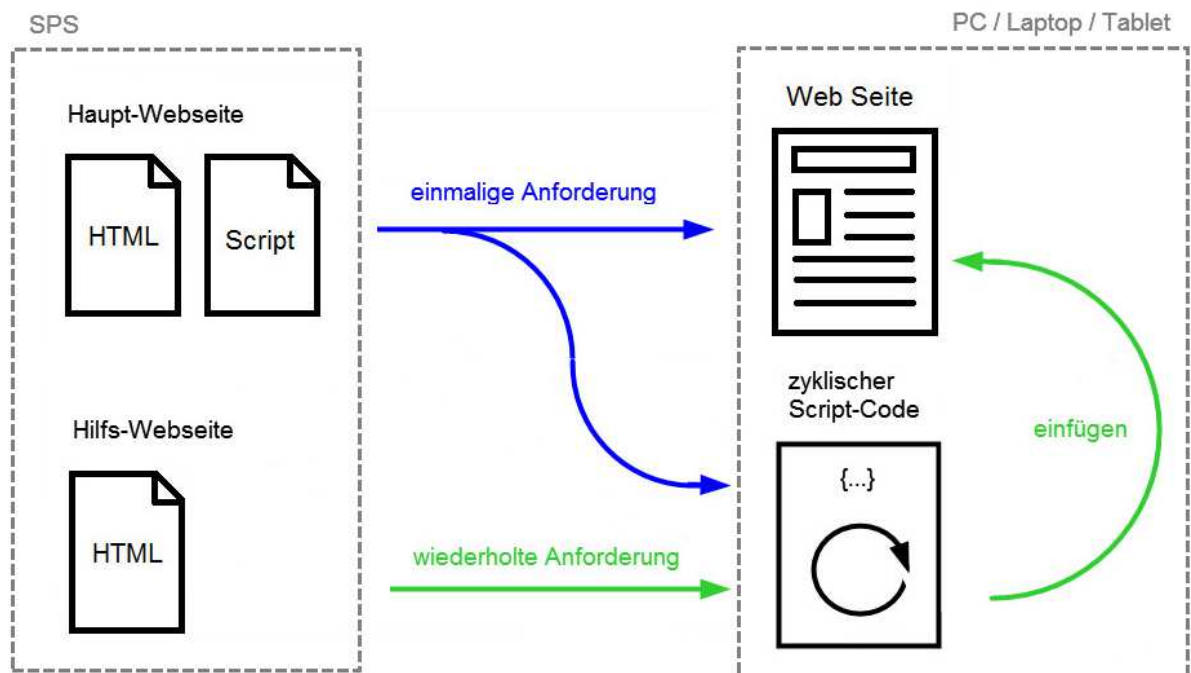


Abbildung 32 - Einmalige und zyklische Aktualisierung

Um den Datenverkehr möglichst gering zu halten wurde eine kleinere Hilfs-Webseite erstellt. Diese enthält nur die aktuellen Variablenwerte. Die Haupt-Webseite samt Script wird also nur einmal geladen. Wird das Script dann im Client ausgeführt, dann fordert es die Hilfs-Webseite zyklisch an. Auch in der Hilfs-Webseite befinden sich AWP-Kommandos, welche bei jeder Anforderung durch aktuelle Variablenwerte ersetzt werden. Diese Werte kann das Script dann in die Haupt-Webseite zur Laufzeit einfügen. In der vorliegenden Anwendung können so Aktualisierungszyklen von weniger als 0,4 Sekunden erreicht werden.

4.3.3.3 SPS-Variablen schreiben

Um Benutzereingaben von der Webseite zur SPS zurück zu schicken verwendet Siemens einen Mechanismus, der schon seit langem ein standardisierter Teil von HTML ist. Es handelt sich um HTML-Formulare, repräsentiert durch das HTML-Element `<form>`. Es

können sowohl die GET-Methode als auch die POST-Methode verwendet werden. Auch Cookies sind nutzbar. Allerdings braucht Web2PLC hier ein weiteres AWP-Kommando.

Hier ein Beispiel:

```
<!-- AWP_In_Variable Name="iTemp" --> AWP-Kommando  
Variable schreiben  
  
<form method="get" action="">  
  Soll-Temperatur:  
  <input name="iTemp" value=":=iTemp:" type="text" size="8">  
    °C  
</form>  
                                zu schreibende  
                                SPS-Variable  
                                AWP-Kommando  
                                Variable lesen (optional)
```

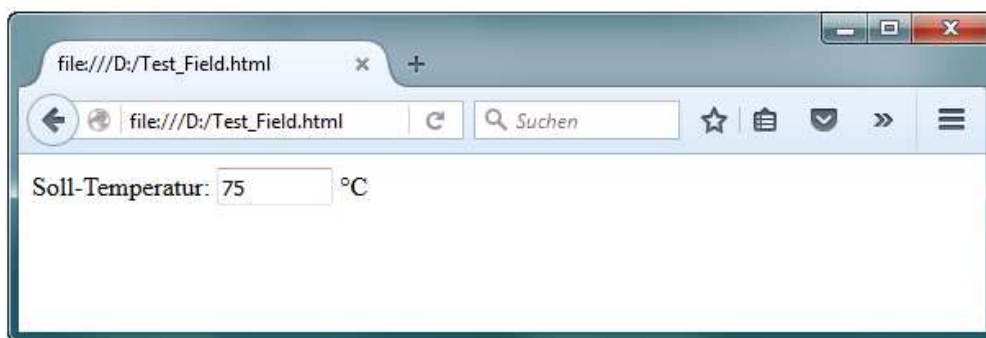


Abbildung 33 - Beispiel SPS-Variable schreiben

Das Bild zeigt oben den Code sowie darunter die entsprechende Darstellung im Webbrowser. Das `<input>`-Element vom Typ „text“ erzeugt ein Eingabefeld. Das `<form>`-Element ist zur Laufzeit unsichtbar und dient zur Gruppierung. In einem `<form>`-Element können mehrere `<input>`-Elemente enthalten sein. So können mehrere Variablen in einer Anforderung gesetzt werden. Das Kommando „AWP_In_Variable“ ist für Web2PLC bei schreibenden Variablen zwingend erforderlich. Es muss aber nicht zwingend an dieser Stelle stehen, es kann z.B. auch an den Anfang des HTML-Dokumentes gesetzt werden. Der Variablenname, hier „iTemp“, muss in der Symbolliste oder in einem Datenbaustein des S7-Projektes angelegt sein.

Das Attribut „value“ ist optional. Es initialisiert das Eingabefeld mit einem Anfangswert. Der Ausdruck „:=iTemp:“ wird beim Laden der Webseite durch den aktuellen Variablenwert ersetzt. Somit kann der Benutzer den aktuell eingestellten Wert ablesen. Nach Eingabe eines neuen Wertes und betätigen der Eingabe-Taste werden die Formular-Werte an den Webserver (bzw. die SPS) zurück geschickt.

Das nächste Beispiel zeigt wie ein Bit durch einen Button gesetzt werden kann. In der Automatisierung ein häufig angewendetes Muster.


```

<!-- AWP_In_Variable Name='bBit' --> AWP-Kommando
<form method="get" action="">
  Maschine starten:
  <input name='bBit' value="1" type="hidden">
  <input value="START" type="submit">
</form>

```

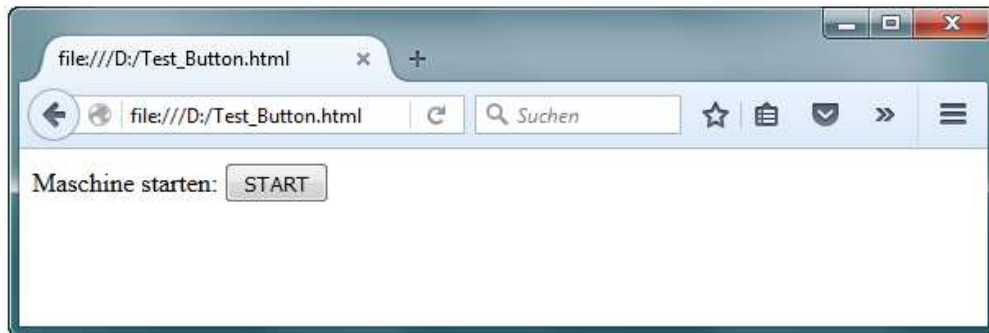


Abbildung 34 - Beispiel Variable setzen über Schaltfläche

Auch hier ist wieder das einleitende AWP-Kommando erforderlich. Jetzt sind allerdings zwei <input>-Elemente innerhalb eines Formulars. Das Erste ist unsichtbar und dient nur dazu den zu schreibenden Wert zu halten. Das Zweite ist vom Typ „submit“ und löst das senden des Formulars aus. Da das Value-Attribut eine 1 beinhaltet wird die Variable „bBit“ in der SPS gesetzt.

Die hier gezeigte Methode mittels Formularen hat den Vorteil, dass sie ohne Script-Programmierung auskommt. Somit ist sie auch auf älteren und einfacheren Geräten lauffähig. Es ist auch möglich Variablen durch Skript-Befehle zu setzen.

4.3.4 Animierte Grafiken

Statische und animierte Bilder (z.B. IMG, GIF) gehören inzwischen zum Standard. Aber auch interaktive Grafiken sind für die meisten Webbrowser kein Problem mehr. Dazu bieten sich u.a. die folgenden beiden Features an.

4.3.4.1 SVG-Element

SVG (Scaleable Vector Graphics) sind Bilder, die aus einem Algorithmus heraus generiert werden. Dabei werden sie aus einzelnen Grafik-Elementen zusammengesetzt, also zur Laufzeit generiert. Die einzelnen Grafik-Elemente sind in XML-Notation geschrieben. Inline-SVG-Elemente können genau wie andere HTML-Elemente direkt in ein Dokument eingefügt werden. Der Code <svg><polyline points="10,10 40,40 10,70"/></svg> erzeugt z.B. eine Linie, die durch drei Stützpunkte verläuft. Ideal um z.B. einen Signalverlauf in einem Diagramm darzustellen. Über JavaScript und DOM-Manipulation kann die Linie zur Laufzeit verändert werden. So kann z.B. das Navigieren im Diagramm durch Ziehen mit der

Maus erreicht werden. Im Rahmen dieses Projektes wurde ein dynamisches SVG-Diagramm zur Visualisierung der Messdaten erstellt. Allerdings tauchen bei der Arbeit mit SVG einige Kritikpunkte auf:

- Ein Inline-SVG-Element benötigt zur Laufzeit mehr Speicher als ein entsprechendes Canvas-Element
- Bei zyklischer Veränderung der Eigenschaften der SVG-Elemente über die DOM-Schnittstelle steigt der Speicherbedarf des Browsers immer weiter an, was auf ein „Memory Leak“ hindeutet.
- Die kontinuierliche Animation von SVG-Elementen läuft deutlich langsamer als bei Canvas-Darstellungen
- In manchen Fällen wird ein SVG-Element in den verschiedenen Browsern unterschiedlich dargestellt. Manche Browser implementieren nicht die volle SVG-Spezifikation.

Aufgrund dieser Nachteile wurde auf das Canvas-Element umgeschwenkt.

4.3.4.2 Canvas-Element

Das HTML-Element `<canvas>` generiert im Gegensatz zu SVG noch keine grafischen Elemente. Stattdessen erzeugt es ein leeres Rechteck. Diese leere Bildschirmfläche muss dann durch Skript-Befehle befüllt werden. Dazu enthält das Canvas-Objekt einige Objekt-Methoden, die der Programmierer verwenden kann. Diese Methoden dienen z.B. zum Zeichnen von Rechtecken, Linien und Punkten. Beim Canvas-Element erzeugen diese Befehle lediglich Pixel-Grafiken, während bei SVG jede Linie ein Objekt mit DOM-Interface darstellt. Die Manipulation von Canvas-Grafiken erfolgt dadurch flüssiger, auch der Speicherbedarf hält sich in Grenzen. Im Rahmen dieses Projektes wird das Canvas-Element verwendet, um Messwerte grafisch darzustellen.

4.3.5 Tools für die Web-Programmierung

Es gibt eine große Anzahl von grafischen Editoren für das Erstellen von Webseiten. Allerdings wird dem Web-Programmierer dadurch trotzdem nicht erspart sich mit der Syntax von HTML, CSS und JavaScript auseinander zu setzen. Die in dieser Arbeit erstellte Benutzeroberfläche ist im Umfang relativ überschaubar, sodass die Erstellung zur Gänze im Texteditor „Notepad++“ erledigt werden konnte. Dieser einfache Editor unterstützt bereits Syntaxhervorhebung, was die Arbeit mit dem Quellcode erheblich übersichtlicher gestaltet.

Für größere Projekte sind grafische Web-Editoren mit Funktionen wie Bibliotheksverwaltung und Code-Autovervollständigung wünschenswert. Aber auch solche Editoren werden in der Regel nicht die oben erwähnten AWP-Kommandos unterstützen. Grafisch aufwändige Steuerelemente, wie z.B. Zeiger-Instrumente und Kurvenanzeigen, sind in HTML nicht vorgesehen und können/müssen durch entsprechende Code-Schnipsel und Frameworks nachgebildet werden. Bei größeren Projekten kann es daher sinnvoll sein ein kom-

merziell erhältliches Web-Framework einzusetzen. Es gibt Softwarehersteller, die dazu eine entsprechend spezialisierte IDE samt Bibliothek liefern.

Zum Testen und Debuggen der Webseite kann der Webbrowser selbst verwendet werden. Die zurzeit üblichen Browser bringen bereits umfangreiche Debugging-Möglichkeiten mit. In vorliegenden Fall wurde der Internet Explorer 10 verwendet, da er auf den meisten Firmenrechnern verwendet wird. Die Vorgehensweise ist einfach: Zuerst wird die zu untersuchende Webseite auf den Webserver der SPS geladen. Dann wird der Browser gestartet, dort kann mit der Taste F12 das Debugging-Fenster geöffnet werden. Jetzt wird die Webseite vom Webserver abgerufen. Tauchen während des Ladens der Seite oder während der Bedienung Fehler auf, so werden eine entsprechende Fehlermeldung sowie die betreffende Code-Stelle im Debugging-Fenster angezeigt. Außerdem sind Tools zur Untersuchung der Performance enthalten, wie etwa der Netzwerk-Profiler. Er zeichnet Netzwerk-Ladezeiten auf und stellt diese grafisch dar.

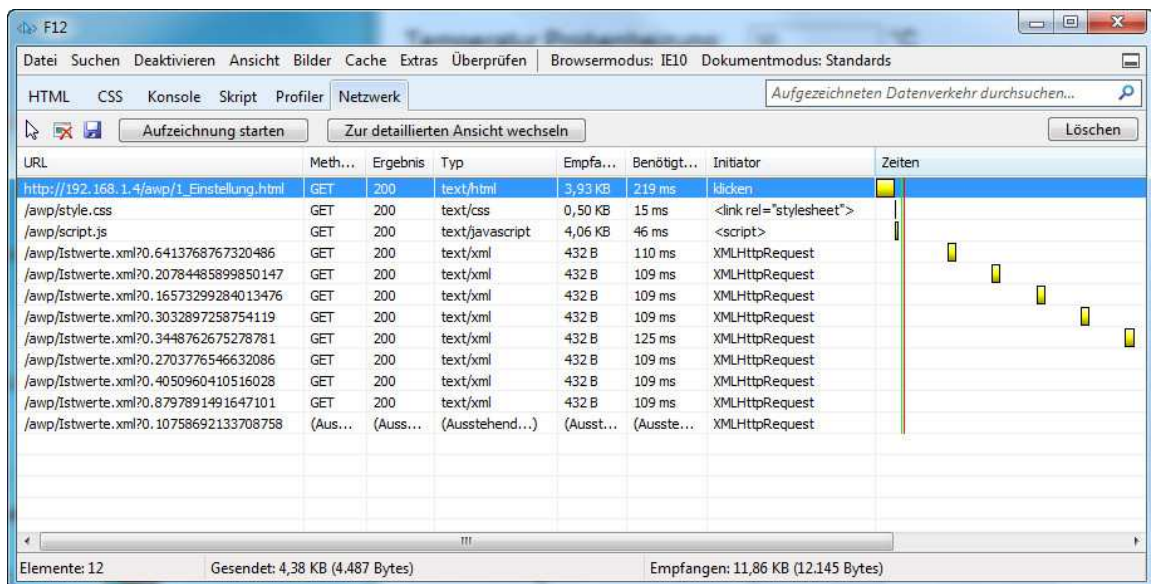


Abbildung 35 - Netzwerk Profiler⁴

Abbildung 35 zeigt beispielhaft wie die Hilfs-Webseite mit den Istwerten der Zugprüfmaschine zyklisch neu angefordert werden. Von der Anfrage bis zur Rückmeldung vergehen ca. 110ms.

⁴ Bildschirmfoto Internet Explorer 10

4.4 Aufbau der Benutzeroberfläche

In diesem Abschnitt werden die einzelnen Seiten der Benutzeroberfläche vorgestellt.

Vor dem Zugriff muss der Benutzer sein Anzeigegerät (PC/Smartphone) mit dem LAN bzw. WLAN der Zugprüfmaschine verbinden. Danach muss der Webbrowser des Gerätes gestartet werden. Die Startseite ist unter der URL „<http://192.168.1.4/awp/Startseite.html>“ zu abzurufen. Bei regelmäßiger Nutzung empfiehlt sich die Erstellung einer Verknüpfung. Da der Webserver der SPS eine Authentifizierung verlangt, wird der Benutzer gegebenenfalls auf die Login-Seite weitergeleitet. Dort muss er sich mit Benutzernamen und Kennwort anmelden.

4.4.1 Startbild

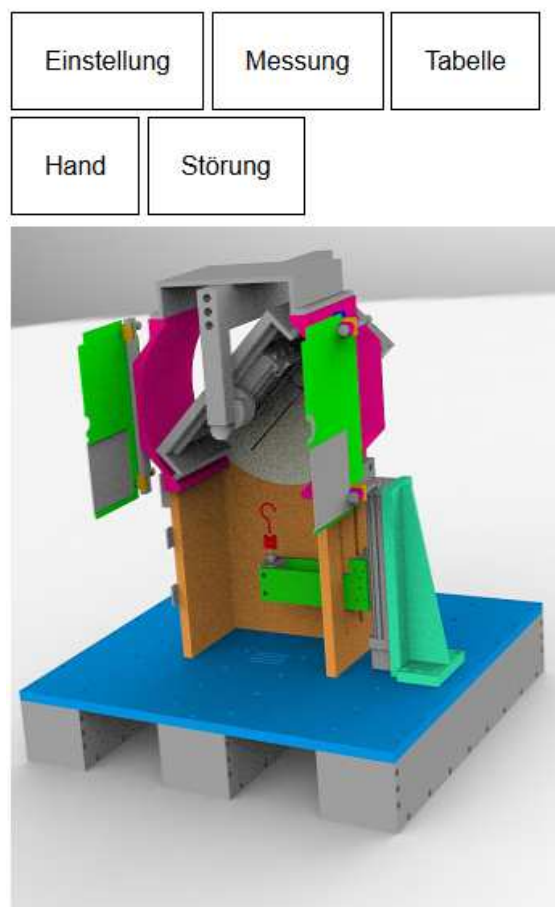


Abbildung 36 - Startbild

Das Startbild dient nur als Einstiegsstelle. Die fünf Schaltflächen im Oberen Bildbereich dienen der Navigation zu anderen Bildschirmseiten. Sie sind in jeder Bildschirmseite zu sehen.

4.4.2 Einstellungen

Auf der Einstellungsseite können alle Parameter zum bevorstehenden Messablauf eingegeben werden. Im oberen Bereich ist wieder die Navigations-Leiste. Knapp darunter wird der aktuelle Messwert des Kraftsensors angezeigt. Danach folgen die Einstellungen.

Einstellung	Messung	Tabelle	Hand	Störung
-------------	---------	---------	------	---------

Zugkraft: 6.49 N

Einstellungen zum Messablauf

Temperatur Probenheizung:	<input type="text" value="30"/>	°C
Temperatur Heißluftheizung:	<input type="text" value="30"/>	°C
Drehzahl Gebläse:	<input type="text" value="50"/>	%
Maximale Zugkraft:	<input type="text" value="10"/>	N
Vorschub Geschwindigkeit:	<input type="text" value="0.2"/>	mm/s
Messzeit:	<input type="text" value="200"/>	s
Messstrecke:	<input type="text" value="40"/>	mm
Entspannstrecke:	<input type="text" value="1"/>	mm
Abtastintervall:	<input type="text" value="1"/>	x100ms
Horizontalantrieb:	<input type="text" value="AUS"/>	
Widerholungen:	<input type="text" value="0"/>	
Zugkraft Erhöhung:	<input type="text" value="0"/>	N
<input type="button" value="übernehmen"/>		

Abbildung 37 - Einstellungen

In der Zeile für die Temperatur Probenheizung lässt sich der Sollwert für den Temperaturregler einstellen. Die Temperatur Heißluftheizung ist der Sollwert für den Temperaturregler des Lufterhitzers. Darunter kann die Drehzahl des Gebläses eingestellt werden. Der Befehl wird per Analogwert direkt an den Controller im Gebläse weitergegeben.

Die Zeile für maximale Zugkraft ist der Sollwert für den Kraftregler. Wird keine Zugkraftbegrenzung gewünscht dann muss hier der maximale für den Sensor verträgliche Wert eingegeben werden.

Bei der Vorschubgeschwindigkeit handelt es sich um die Geschwindigkeit mit der die Deckschicht von der Probe abgezogen wird, sofern sie nicht von der Kraftbegrenzung gedrosselt wird. Danach folgen die Eingabefelder für Messzeit und Messstrecke. Eine Messung ist abgeschlossen sobald entweder die Messzeit abgelaufen ist oder die Messstrecke erreicht wurde. So können beide Anwendungsfälle abgedeckt werden.

1. Messung der maximalen Abreißkräfte: Die maximale Zugkraft und die Messzeit werden sehr hoch eingestellt, so dass sie voraussichtlich nicht erreicht werden. Die Maschine fährt bis zur Eingestellten Messstrecke mit der gewünschten Vorschubgeschwindigkeit. Die minimale und maximale Abreißkraft kann aus dem Diagramm entnommen werden.
2. Messung der Kriechgeschwindigkeit: Die maximale Zugkraft wird auf den zu untersuchenden Wert herabgesetzt. Die Maschine drosselt die Vorschubgeschwindigkeit damit die Zugkraft nicht überschritten wird, es stellt sich ein Gleichgewicht in der Höhe der Kriechgeschwindigkeit ein. Da diese Geschwindigkeiten und Wege sehr gering sind wird die Messzeit zum begrenzenden Faktor.

Nach der Messung fährt die Vertikalachse wieder ein Stück zurück um die Probe zu entspannen. Dieser Weg kann in der Einstellung „Entspannstrecke“ festgelegt werden. Als nächstes folgt die Einstellung des Abtastintervalls für Langzeitmessungen. Mit der Auswahl „Horizontalantrieb“ kann der Horizontalantrieb deaktiviert werden. Darunter kann die Anzahl der Mess-Wiederholungen eingestellt werden. Der Wert der „Zugkraft Erhöhung“ wird bei jeder Wiederholung auf die maximale Zugkraft aufaddiert.

4.4.3 Messung, Istwerte

Nachdem alle Parameter eingestellt wurden kann die Messung gestartet werden. Die Starttaste befindet sich am unteren Rand der folgenden Bildschirmseite.

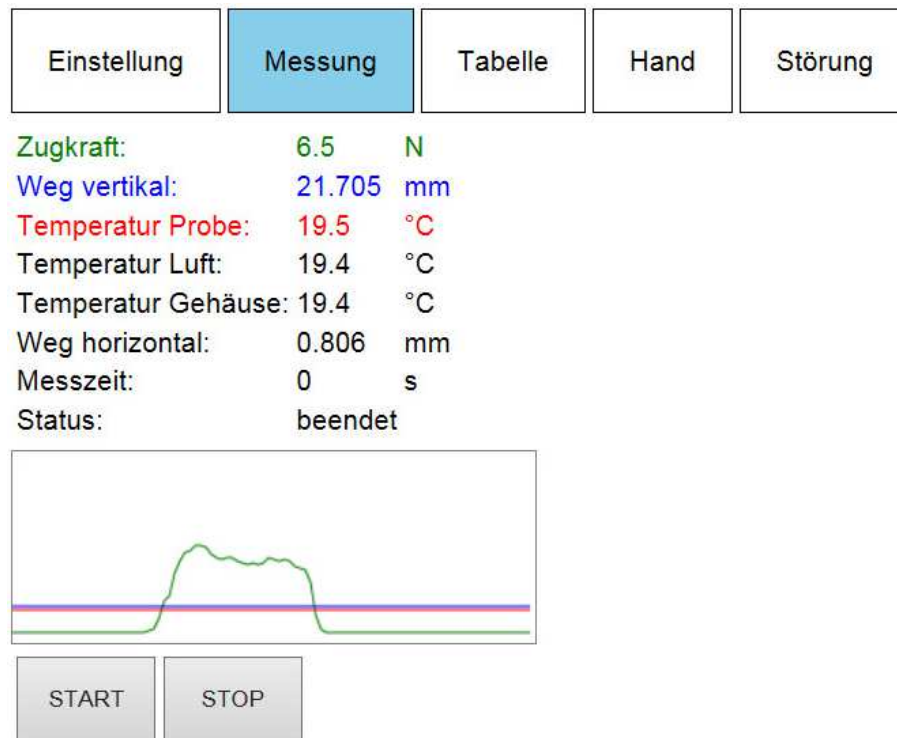


Abbildung 38 - Messung

Auf dieser Seite kann der Benutzer die aktuelle Messung verfolgen. Unter der Navigationsleiste werden alle aktuellen Sensorwerte angezeigt. Die letzte Zeile zeigt den Status der laufenden Messung. Der Status zeigt an, in welchem Schritt sich die Schrittkette der SPS gerade befindet. Ist die Messung beendet dann erscheint eine entsprechende Statusmeldung.

Am unteren Rand des Bildschirms befinden sich die Schaltflächen zum Starten und Stoppen der Messung. Die Stopptaste sorgt für ein frühzeitiges abbrechen einer Messung. Im Regelfall stoppt die Messung automatisch.

Dazwischen befindet sich ein einfaches Liniendiagramm. Es dient lediglich als Vorschau bzw. zur anschaulichen Darstellung der Messwerte. Das Diagramm läuft endlos. Die tatsächlichen Messwerte können auf der nächsten Seite abgerufen werden.

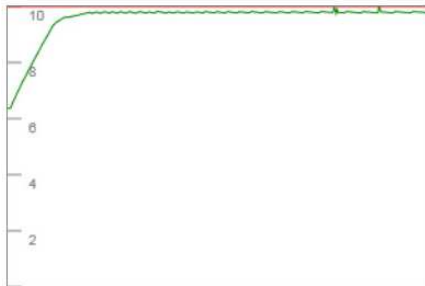
4.4.4 Ausgabe in Tabellenformat

Während einer Messung werden die Messwerte in der SPS spannungsausfallsicher gespeichert. Auf dieser Seite können sie abgerufen werden.

Einstellung	Messung	Tabelle	Hand	Störung
-------------	---------	----------------	------	---------

Zugkraft: -0.52 N

Letzte Messung



Messwerte

Kraft	[N]	6,45	6,47	6,55	6,64	6,73	6,8	6,88	6,96	7,04	7,11	7,18	7,26	7,34	7,41	7,48
Weg vertikal	[mm]	0,004	0,024	0,044	0,064	0,084	0,105	0,124	0,144	0,164	0,184	0,204	0,224	0,245	0,264	0,284
Weg horizontal	[mm]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Temperatur Probenheizung	[°C]	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5	19,5

Abbildung 39 - Tabelle

Im unteren Bildschirmbereich befindet sich die Tabelle mit den zuletzt aufgezeichneten Messwerten. Die Tabelle enthält vier Zeilen mit je zweitausend Einträgen. Jede Spalte entspricht einem Abtastintervall. Das kleinste einstellbare Abtastintervall beträgt einhundert Millisekunden. Durch den horizontalen Schieberegler kann der Benutzer alle Messwerte durchblättern. In der Regel werden die Messwerte aber in einem Tabellenkalkulationsprogramm wie z.B. MS-Excel weiter verarbeitet. Dazu kann die gesamte Tabelle einfach per Copy-Paste-Funktion kopiert und direkt in MS-Excel eingefügt werden.

In der Mitte des Bildes werden die Messwerte grafisch dargestellt. Das Diagramm wird erst nach dem Laden der Seite direkt aus den Tabellenwerten generiert. Das lässt sich über JavaScript sehr gut lösen. Falls der Browser kein JavaScript unterstützt bleibt das Diagramm leer. In dem Fall erhält der Benutzer nur die Messwerttabelle.

Als Nachteil dieser Methode ist die sehr lange Ladezeit der Tabelle zu erwähnen. Interessanterweise werden Bilddateien, welche ebenso viel Speicher benötigen, viel schneller geladen als die Tabelle. Die Verzögerung entsteht vermutlich durch das Parsen der AWP-

Kommandos, welche bei der Tabelle in sehr großer Zahl vorkommen. Es handelt sich also um eine Eigenheit des Webservers.

4.4.5 Handbetrieb

Diese Seite dient zum manuellen Ausrichten der Achsen.

Einstellung	Messung	Tabelle	Hand	Störung
-------------	---------	---------	------	---------

Zugkraft: -0.52 N

Antrieb horizontal

LINKS	STOP	RECHTS
-------	------	--------

Antrieb vertikal

AUF	STOP	AB	SPANNEN
-----	------	----	---------

Heißluftheizung

EIN	AUS
-----	-----

Probenheizung

EIN	AUS
-----	-----

Abbildung 40 - Handbetrieb

Durch betätigen der Schaltflächen setzt sich die jeweilige Achse in Bewegung. Die Schaltfläche STOP hält sie wieder an. Die manuelle Steuerung ist notwendig, da die Proben unterschiedlich lang sein können bzw. an unterschiedlichen Stellen im Horizontalantrieb eingespannt werden können. Mit dieser Funktion kann der Benutzer die Achsen so Ausrichten, dass die Probe immer in der Mitte der Messkammer sitzt.

Außerdem kann der Benutzer entscheiden, ob die Probenheizung oder die Heißluftheizung mit verwendet werden sollen oder nicht. Nach dem Einschalten der Maschine sind die Heizungen deaktiviert. Erst nach Aktivierung durch den Button startet die Regelungs-Software und der Aufheizvorgang beginnt.

Eine neu eingespannte Probe sitzt in der Regel noch locker, sodass am Beginn einer Messung ein Leer-Weg entsteht. Mit der Taste „Spannen“ kann dieser Leer-Weg bereits vor der Messung überbrückt werden. Dabei fährt der Horizontalantrieb nach unten und stoppt automatisch wenn der Kraftsensor 1N überschreitet.

4.4.6 Störungsanzeige

Wird eine Fehlermeldung ausgelöst, dann beginnt die Schaltfläche „Störung“ in der Navigationsleiste zu blinken. Ein Klick auf die Schaltfläche öffnet die Störmeldungsanzeige.



Abbildung 41 - Störmeldungsanzeige

Hier werden alle anstehenden Meldungen aufgelistet. Jede Meldung setzt sich zusammen aus Störmeldenummer und Meldetext. Wurde der Fehler behoben dann kann mit der Taste „Quittieren“ der Fehlerzustand beendet werden.

4.4.7 Justieren des Kraftsensors

Diese Seite dient zum Justieren des Kraftsensors. Die Justage ist nur notwendig wenn der Kraftsensor ausgetauscht wurde.

Einstellung	Messung	Tabelle
Hand	Störung	

Zugkraft: -0.59 N

Kraftsensor justieren

1. Justagegewicht eingeben:

1/100 N

2. Nullpunkt einlernen

3. Justagegewicht einlernen

Abbildung 42 - Justieren

Es wird vorausgesetzt, dass der Sensor eine lineare Kennlinie besitzt. Als erster Kennlinien-Punkt wurde der Nullpunkt festgelegt. Der zweite Kennlinienpunkt ist frei wählbar, sollte aber möglichst im oberen Ende des Sensor-Nennbereiches liegen. Die Taste „Nullpunkt einlernen“ muss betätigt werden während der Sensor frei von allen Kräften ist. Danach wird der Sensor mit einer bekannten Kraft belastet, dem Justagegewicht. Der genaue Wert muss in das Eingabefeld eingetragen werden. Mit der Schaltfläche „Justagegewicht einlernen“ wird der Justiervorgang abgeschlossen.

5 Ergebnis

5.1 Zusammenfassung

Für die in Kapitel 1 dargestellte Zugprüfmaschine wurde die passende Steuerungstechnik entworfen. Die Software wurde speziell an die Anforderungen der F&E angepasst.

Die zentrale Steuerung vom Typ Siemens IM151-8 F CPU ist für die Aufgabe gut geeignet. Sowohl Speicher als auch Rechenleistung sind ausreichend.

Bei den Linearachsen mussten auf Grund der geringen Soll-Geschwindigkeiten zusätzliche Getriebe zwischengeschaltet werden. Durch das hohe Übersetzungsverhältnis kam es zu folgenden Problemen:

- Der Zähler des Drehgebers am Antriebsmotor erreicht den Überlauf, sodass nicht der volle Fahrweg ausgenutzt werden kann. Als Abhilfe werden die Achsen vor jedem Messvorgang auf Null gesetzt. Zusätzlich wird der Fahrweg durch Hardwareendschalter begrenzt.
- Das Drehmoment der Achse wird durch das Getriebe dermaßen hoch übersetzt, dass die Kraftregelung des Motorcontrollers nicht genutzt werden kann. Stattdessen wurde eine eigene Regelschleife über den Kraftsensor und über das SPS-Programm aufgebaut. Dadurch wird die Ansteuerung des Servocontrollers etwas komplexer.

Die Probenheizung wird über einen Softwareregler angesteuert. Die Leistungsstellung erfolgt über ein SSR. Trotz der relativ großen Heizleistung konnte ein gutes Regelergebnis erzielt werden.

Das Heißluftgebläse ist für diesen Einsatzzweck nicht ganz optimal, da es für weitaus höhere Temperaturen ausgelegt ist. Dadurch kann die im Gerät integrierte Temperaturregelung nicht genutzt werden. Durch die Umstellung auf „Open Loop“ und den Aufbau einer eigenen Regelschleife konnte das Ergebnis verbessert werden.

Die Ansteuerung des SPS-Eingangsmodule (Typ Siwarex CS) für den Kraftsensor ist vergleichsweise aufwendig aber zweckmäßig. Es wurde ein Algorithmus für die Abreißerkennung entwickelt. Diese funktioniert bei Messung mit höheren Geschwindigkeiten sehr gut, hat jedoch Schwächen bei Messungen mit geringen Geschwindigkeiten (Kriech-Effekt bei Kunststoffen). Die Aufzeichnung der Messwerte läuft mit dem eigens entwickelten Datenlogger-Baustein problemlos. Allerdings mussten auf Grund des begrenzten Speichers Kompromisse bei Langzeitmessungen gefunden werden.

Die Visualisierung mittels Webbrowser funktioniert überraschend gut. Es wird eine hohe Plattformunabhängigkeit erreicht. So kann die Zugprüfmaschine u.a. auch mit dem Smartphone gesteuert werden, was unter den Nutzern für Begeisterung sorgt. Jedoch ist für das Erstellen einer solchen Visualisierung eine Einarbeitung in die Webprogrammierung notwendig. Zur graphischen Darstellung der Messwerte wurde das HTML-Element „Canvas“ verwendet. Es zeigt eine bessere Performance als das SVG-Element. Die zyklische Aktualisierung der Messwerte gelingt mittels JavaScript sehr gut. Etwas unangenehm ist die lange Ladezeit der Messwerte-Tabelle, welche auf das Umsetzen der AWP-Kommandos zurückzuführen sind.

5.2 Ausblick

Die Zugprüfmaschine wird natürlich auch in Zukunft weiter gepflegt und auf neue Bedürfnisse angepasst. Verbesserungsmöglichkeiten sind:

- Verbesserung des Algorithmus der Abreißerkennung, sodass auch die Einflüsse des Kriech-Effektes verarbeitet werden können.
- Verbesserung der Ladezeiten der Messwertetabelle. Das könnte eventuell erreicht werden indem der Datenlogger direkt in die Datenbausteine der Webseite schreibt. Das würde das AWP-Parsing an dieser Stelle überflüssig machen.
- Mittels JavaScript ist es möglich Diagramme interaktiv zu gestalten. Auch der Seitenwechsel durch Wisch-Gesten wäre machbar. So könnte das „Look and Feel“ der Benutzeroberfläche wesentlich verbessert werden.

5.3 Fazit

Die Zugprüfmaschine liefert der Firma Frimo bereits wertvolle Daten für die Prozessoptimierung. Die Einbindung von Smartphones und Tablets spiegelt den Zeitgeist wieder und wird auch gerne für Demonstrationszwecke verwendet.

Literatur

- [DinKo2014] Dines, Daniel; Kocher, Andreas: Entwicklung eines Zugversuchsprüfstandes für Klebeverbindungen – 2014 – Salzburg, HTBLuVA, Anlagenbau, Diplomarbeit, 2014
- [FrHoe2013] Hörzelberger, Fa. Frimo Freilassing GmbH, Schulungsunterlagen Nutkaschieren, Präsentationsunterlagen, 2013
- [FrFrau2013] Frauenschuh, Fa. Frimo Freilassing GmbH, Zugversuchsgerät, Präsentationsunterlagen Lastenheft, 2013
- [FesSp2015] Fa. Festo AG & Co. KG, Spindelachsen EGC-BS-KF mit Kugelumlauflführung, Auswahlhilfe, 2015, verfügbar unter <https://www.festo.com/>
- [FesBa2015] Fa. Festo AG & Co. KG, Spindelachse EGC-BS-KF, Bedienungsanleitung, 2015, verfügbar unter <https://www.festo.com/>
- [Hotset2013] Fa. HotSet Heizpatronen und Zubehör GmbH, Infolyer Hotflex flexibler Rohrheizkörper, 2013, verfügbar unter <http://www.hotset.com/>
- [Leister2013] Fa. Leister Technologies AG, Hotwind System, Bedienungsanleitung, 2013, verfügbar unter <https://www.leister.com/de-bz/process-heat/produkte/heissluftgeblaese/hotwind-system>
- [Me2014] Fa. ME-Meßsysteme GmbH, Kraftsensor KD24s, Datenblatt, 2014, verfügbar unter <http://www.me-systeme.de/sensorik/kraftsensoren/kd24s.html>
- [Jum2015] Jumo GmbH & Co. KG: Aufbau und Anwendung von Platin Temperatursensoren, verfügbar unter <http://www.jumo.de/>, Zugriff am 20.12.2015
- [Siwa2011] Fa. Siemens AG, Siwarex CS, Gerätehandbuch, 2011, verfügbar unter <https://support.industry.siemens.com>

- [WelZas1998] Wellenreuther, Günter; Zastrow, Dieter: Steuerungstechnik mit SPS – 5. Aufl. Braunschweig/Wiesbaden: Vieweg, 1998
- [Bon2014] Bonnet, Martin: Kunststofftechnik - 2. Aufl. Springer Vieweg, 2014
- [Fhpp2009] Fa. Festo AG & Co. KG: Festo Handling an Positioning Protokoll, Beschreibung, Ausgabe de 1006a, verfügbar unter <https://www.festo.com/>
- [Thk2006] Fa. THK CO. LTD: Kugelgewindetriebe, Merkmale und Typen, Dok. Nr. 502G, verfügbar unter https://tech.thk.com/de/products/pdf/de_b15_006.pdf#1
- [SiePid2003] Fa. Siemens AG: PID Temperature Control, Handbuch, 2003 verfügbar unter <https://support.industry.siemens.com/cs/document/18652438/pid-temperature-control?dti=0&lc=de-DE>
- [Web2015] Online-Dokumentation: <http://www.w3schools.com/>
Zugriff am 20.12.2015
- [Sel2015] Online-Dokumentation: <https://selfhtml.org/>
Zugriff am 20.12.2015
- [Ack2015] Ackermann, Philip: Professionell entwickeln mit JavaScript – 1.Auflage, Bonn: Rheinwerk Computing, 2015
- [W3C] World Wide Web Consortium, <http://www.w3.org/TR/2014/REC-html5-20141028/>, Zugriff am 20.12.2015
- [Par2008] Parthier, Rainer: Messtechnik, 4. Auflage, Wiesbaden: Vieweg, 2008

Anlagen, Teil 1: Technische Daten Kraftsensor

Technische Daten des Kraftsensors KD24s

Maße / Material		
Material		Aluminium-Legierung
Abmessungen		24 x 26 x 10
Krafteinleitung / Gewinde		2x M5x0,8
mechanische Daten		
Nennkraft(FS)	N	± 2, 10, 20, 50; 100, 200, 500, 1000
Gebrauchskraft	%FS	400% ≤50N, 200% ≤100N, 150% ≥500N
Bruchkraft	%FS	600% ≤50N, 400% ≤100N, 300% ≥500N
Messweg bei FS	mm	0,05...0,1
elektrische Daten		
Nennkennwert 1)	mV/V@FS	0,5
Nullsignaltoleranz	mV/V	0,05
max. Speisespannung	V	5
Eingangswiderstand	Ohm	390±40
Ausgangswiderstand	Ohm	350±1,5
Isolationswiderstand	MOhm	>5 10 ⁹
Anschlusskabel STC-31V-4RWBG	m	3
Genauigkeit		
Genauigkeitsklasse	%	0,1
rel. Linearitätsabweichung	%FS	0,02
rel. Umkehrspanne	%FS	0,02
Temperaturkoeffizient des Nullsignals	%FS/K	0,02
Temperaturkoeffizient des Kennwertes	%RD/K	0,01
Kriechfehler (30 min)	%FS	0,1
Temperatur		
Nenntemperaturbereich	°C	-10... +70
Gebrauchstemperaturbereich	°C	-10 ... +85
Lagertemperaturbereich	°C	-10 ... +85
Schutzart		IP65

Abkürzungen: RD: Istwert („Reading“); FS: Endwert („Full Scale“);

1) Der exakte Kennwert wird im Prüfprotokoll ausgewiesen.; Druckbelastung: positives Ausgangssignal

Quelle: [Me2014]

Anlagen, Teil 2: Technische Daten Siwarex

Technische Daten Analogeingangsmodule SIWAREX CS:

EG-Bauartzulassung als nichtselbsttätige Waage, Handelsklasse III	2000 d ($\pi = 0,4$)
Genauigkeit mit Ex-i-Interface	2000 d ($\pi = 0,5$)
Fehlergrenze nach DIN1319-1 vom Messbereichsendwert bei 20 °C ± 10 K	≤ 0,05 %
Aktualisierungsrate intern/extern	50 Hz
Auflösung intern	65.535 Teile
3 Messbereiche	0 bis 1 mV/V 0 bis 2 mV/V 0 bis 4 mV/V
max. Entfernung der WZ (eichfähig)	1000 m (500 m)**
max. Entfernung zwischen WZ und Ex-i-Interface im Ex-Bereich	150/500 m für Gasgruppe IIC 1000 m für Gasgruppe IIB (siehe Gerätehandbuch SIWAREX IS)
kleinstes zulässiges Eingangssignal für einen Eichwert im eichfähigen Betrieb	≥ 1,5 µV/e
Wägezellenspeisung	
Spannung	typ. DC 6 V *
Maximaler Strom	68 mA
zulässiger Wägezellenwiderstand ohne Ex-i-	Eingangswiderstand > 87 Ω

Quelle: [Siwa2011]

Anlagen, Teil 3: Befehlscodes Siwarex CS

Code	Bedeutung des Befehls	Bemerkung
	Service und Justagebefehle	
1	DS4 im Flash sichern Grenzwertparameter im Flash sichern (DS4)	
2	DS26 im Flash sichern Tara- und Nullstellwert im Flash sichern (DS26)	
3	Justagebefehl Nullpunkt gültig Befehl Nullpunkt gültig (aktiviert den Justierbetrieb)	Ausführbar wenn kein Schreibschutz (Drahtbrücke 1-5)
4	Justagebefehl Justagegewicht 1 gültig Der erste Justagepunkt wird durch das Eichgewicht bestimmt.	Ausführbar wenn kein Schreibschutz (Drahtbrücke 1-5)

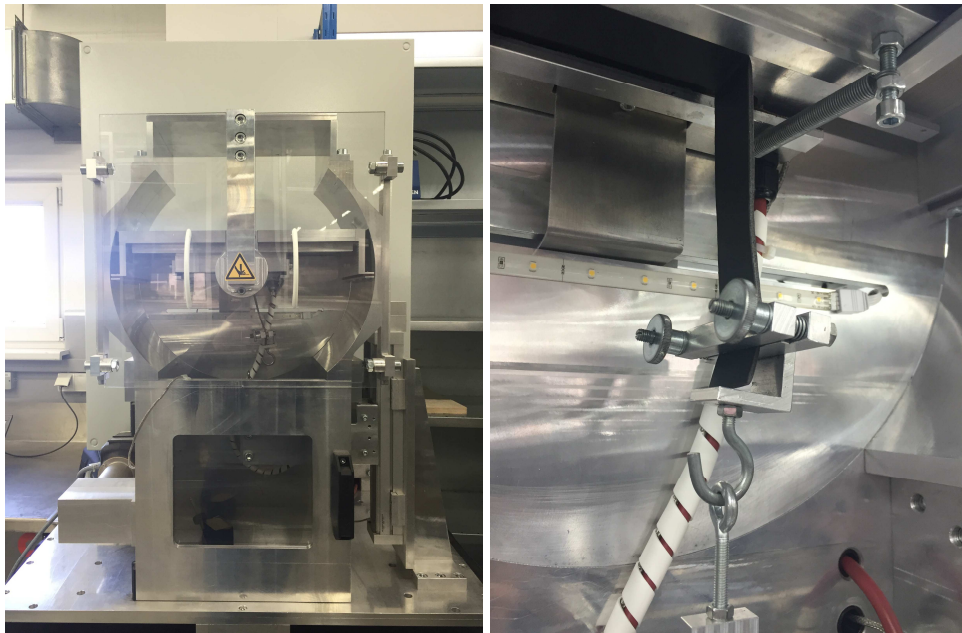
Code	Bedeutung des Befehls	Bemerkung
5	Justagebefehl Justagegewicht 2 gültig Der zweite Justagepunkt wird durch das zweite Eichgewicht bestimmt.	Ausführbar wenn kein Schreibschutz (Drahtbrücke 1-5)
8	Alle Datensätze mit Defaultwerten besetzen Alle Parameter werden auf den Zustand ab Werk vorbesetzt.	Ausführbar wenn kein Schreibschutz (Drahtbrücke 1-5)
15	Kennlinienverschiebung Der Befehl verschiebt die gesamte Kennlinie (die Justagepunkte) so, dass der aktuelle gefilterte Digit-Wert aus dem ADC zu einem neuen Nullpunkt wird. Ergibt sich bei der Verschiebung ein unzulässiger Digit-Wert für einen Justagepunkt, so wird der Befehl abgewiesen.	Ausführbar wenn kein Schreibschutz (Drahtbrücke 1-5)

	Wägebefehle	
21	Waage Nullstellen Das aktuelle Gewicht wird zu Null gesetzt. Im eichfähigen Betrieb („OIML“) nur eingeschränkt (z.B. –1%, +3%) um den Nullpunkt möglich. Gleichzeitig wird das Taragewicht gelöscht.	
22	Tarieren Das aktuelle Gewicht wird zu Null gesetzt, gleichzeitig wird die Gewichtsanzeige als „Netto“ gekennzeichnet.	
23	Tara löschen Das Taragewicht wird gelöscht. Das aktuelle Gewicht wird angezeigt und Kennzeichnung von „Netto“ auf „Brutto“ umgestellt, die Kennzeichnung „Netto“ und ggf. der Status „Preset-Tara“ wird zurückgesetzt.	
24	Eingegebenes Tara aus DS15 übernehmen Das eingegebene Taragewicht wird als Tara übernommen, gleichzeitig wird die Gewichtsanzeige mit „Netto“ gekennzeichnet und der Status „Preset Tara pT“ gesetzt.	
25	Erhöhte Auflösung einschalten Ausgabe/ Anzeige des eichfähigen Gewichtswertes mit erhöhter Auflösung für 5 sec aktivieren.	
26	Taragewicht aus S7-Steuerbereich übernehmen	

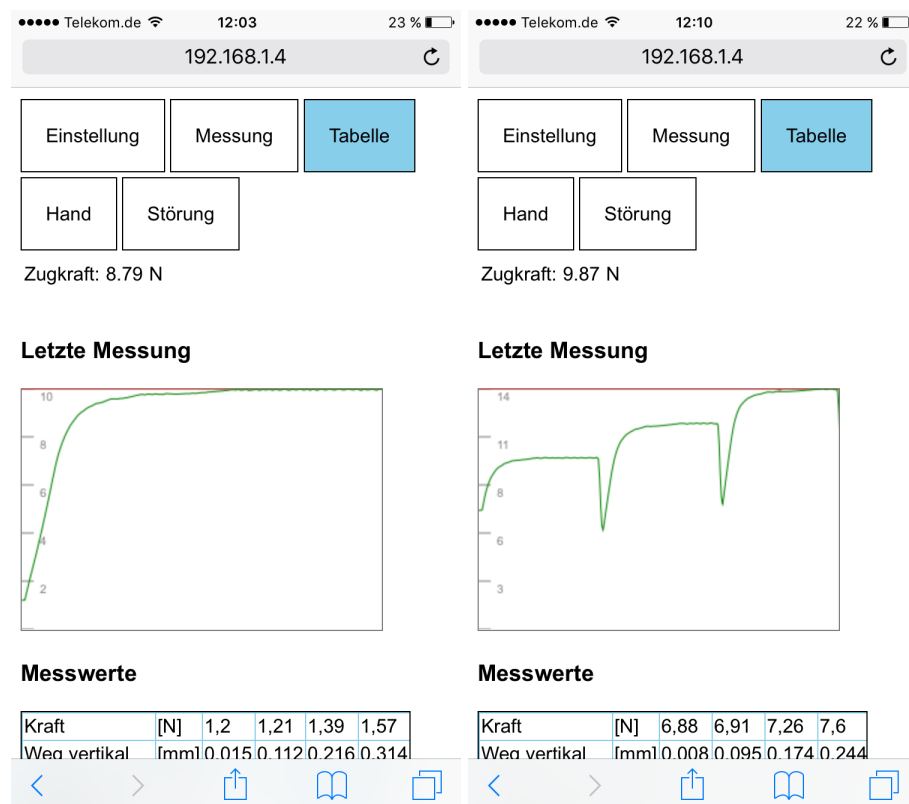
Befehlsgruppe	Beschreibung
1... 99	Befehle werden ohne Lesen oder Schreiben von Datensätzen an die Baugruppe weitergegeben (Waagen-, Wäge-, Protokollbefehle). Die Bedeutung der Befehle entspricht der Aufstellung <i>in Tabelle 6-1 Befehlsliste der SIWAREX CS</i> . Zulässig sind die Nummern für existierende Befehle.
203... 230	Lesen eines Datensatzes 3... 30. Zulässig sind die Nummern für existierende Datensätze.
403... 426	Schreiben eines Datensatzes 3... 26. Zulässig sind die Nummern für existierende Datensätze.
605	Lese alle Datensätze in die SIWAREX CS (DS3, DS4, DS9, DS15, DS30)
610	Schreibe die Datensätze DS3, DS4, DS15 in die SIWAREX CS

Quelle: [Siwa2011]

Anlagen, Teil 4: Bilder der Zugprüfmaschine



Links: die fertige Zugprüfmaschine, rechts: Einspannen der Probe



Links: einfache Messung, rechts: Mehrfachmessung mit 3 Wiederholungen

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Herbert Dirnberger